Protected Area Tools (PAT)
for ArcGIS 9.3™
Version 3.0

USER MANUAL and TUTORIAL

Written by

Steve Schill and George Raber

Funded by

The Inter-American Biodiversity Information Network (IABIN) and

The World Bank Development Grant Facility (DGF)

August 2009

Table of Contents

Acknowledgements 3
Introduction 4
Mandatory Requirements Needed to Run the Protected Area Tools 5
Installing the Protected Area Tools (PAT) v.3 7
The Protected Area GAP Assessment Process 7
MODULE 1: Environmental Risk Surface (ERS) 10
1.0.1 Intensity Value 12
1.0.2 Influence Distance 12
1.0.3 Distance Decay Types 12
EXERCISE 1 Building an Environmental Risk Surface 14
Sample Data: Tydixton Park Watershed, Jamaica 14
1.1 Setting up the ERS Table 15
1.1.1 Overlay Function 17
1.2 Adding and Assigning the Intensity and Influence Distance to Each Risk Element Feature 18
1.3 Executing the ERS Module 18
1.3.1 Launch the ERS Module and Specify Input Layers and Output Parameters 19
1.3.2 Specify the Intensity, Influence Distance, Decay Type, Overlay, and Weight for each Risk Element 22
1.3.3 Viewing the ERS Results and Summarizing Spatial Statistics 24
1.3.4 Creating Freshwater and Marine Environmental Risk Surfaces 26
MODULE 2. Relative Biodiversity Index (RBI) Calculator 31
EXERCISE 2. Calculating the Relative Biodiversity Index (RBI) 34
2.1 Running the RBI Module 35
2.2 Interpreting RBI Results 37
MODULE 3. Marxan Tools 42
3.0 The MARXAN Algorithm 43
EXERCISE 3: Running a Sample Marxan Analysis 44
3.2 Marxan Input File Preparation 49
3.2.1 Marxan Input Files 51
3.2.2 Generating Hexagons (HexGen) 55
3.2.3 Marxan Target Prep 56
3.2.3 Marxan Input Generator (MIG) 61
3.2.4 Combining Marxan Input Files 65
3.2.5 Run Convert to Matrix File 65
3.3 Setting up the input.dat file with Inedit and Executing Marxan 66
3.3.1 Running Inedit 66
3.3.2 Running Marxan 70
3.4 Joining and Displaying Marxan Output 71
3.4.2 Joining Marxan Output Files 71
3.4.2 Displaying and Analyzing Marxan Results 72
References 74

Acknowledgements

The idea for the development of the Protected Area Tools (PAT) to assist countries struggling with methods for conducting national protected area gap assessments was conceived at the Mesoamerica and Caribbean Geospatial Alliance meeting in Port of Spain, Trinidad in May 2004. The design and development of the system was executed by The Nature Conservancy's Mesoamerica & Caribbean Science Program and funded by The World Bank's Development Grant Facility (DGF) Project. The DGF Project was designed to help the Inter-American Biodiversity Information Network (IABIN) establish the Connectivity Program, whose main objective is to encourage the integration of biological and geospatial data. The objective of IABIN is to promote sustainable development and the conservation and sustainable use of biological diversity in the Americas through better management of biological information. This project has established partnerships with organizations or programs in the region with similar goals and which have provided co-financing to meet the specific objectives of the Connectivity Program. In particular we would like to acknowledge Vince Abreu, Douglas Graham, Ivan Valdespino, Boris Ramirez, Rodrigo Tarte, and Sandra Ann Icaza for their help and support offered over the course of this project.

It is hoped that the Protected Area Tools will continue to be developed and refined through external funding support. A variety of new features have been added to version 3 that we hope you will find useful. If you have bugs to report or recommendation to make, please send your comments to the authors:

Steven R. Schill, Ph.D.
Senior Geospatial Scientist
The Nature Conservancy
Mesoamerica & Caribbean Region
sschill@tnc.org

George T. Raber, Ph.D.
Assistant Professor
Department of Geography and Geology
School of Ocean and Earth Sciences
University of Southern Mississippi
graber@gmail.com

DISCLAIMER

The authors would like to expressly state that these scripts are being placed in the public domain or is "Freeware" and may be freely used.

THESE SCRIPTS ARE PROVIDED "AS-IS," WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, BY STATUTE OR OTHERWISE, INCLUDING, BUT NOT LIMITED TO ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHORS DO NOT WARRANT THAT THE OPERATION OF THESE SCRIPTS SHALL BE UNINTERRUPTED OR ERROR FREE. THE USER BEARS ALL RISK AS TO THE QUALITY AND PERFORMANCE OF THESE SCRIPTS.

IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR COSTS OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOST SALES OR BUSINESS EXPENDITURES, INVESTMENTS, OR COMMITMENTS IN CONNECTION WITH ANY BUSINESS, LOSS OF ANY GOODWILL, OR FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT OR USE OF THESE SCRIPTS, HOWEVER CAUSED, ON ANY THEORY OF LIABILITY, AND WHETHER OR NOT THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THESE LIMITATIONS SHALL APPLY NOTWITHSTANDING ANY FAILURE OF ESSENTIAL PURPOSE OF ANY LIMITED REMEDY.

Introduction

Many countries are seeking technical assistance to meet the requirements laid down in the Seventh Conference of the Parties (COP-7) Global Program of Work (PoW) on Protected Areas (PAs). The PoW mandates an established global network of representative and effectively managed national and regional PAs on land by 2010 and at sea by 2012 (CBD, 2001). The Nature Conservancy (TNC) has a vested interest in helping countries develop science-based PA networks and has pledged to build capacity through the development of country-driven National Implementation Support Programs (NISPs) a program that supports the Convention on Biological Diversity.

One way to help overcome the technical challenges of the daunting process of evaluating and filling protected area gaps is the development and use of GIS-based user-friendly tools that support the protected area gap process. The development of a Protected Area Gap Decision Support System (DSS) was conceived as part of an ongoing process to help fill the technical void that exists in many countries. The development of these tools was funded by the Interamerican Biodiversity Information Network (IABIN) and The World Bank Development Grant Facility (DGF). It is part of a process to further support IABIN's objectives and help conservation planners throughout Mesoamerica and the Caribbean assess current PA status and establish priorities for future conservation management. Version 1.0 of these tools was completed in September 2006 and works in ArcGIS 9.1. The new version 3.0 was completed in August 2009, renamed to the Protected Area Tools (PAT) and is now compatible with ArcGIS 9.3. This version includes many new features requested from our users. It is hoped that PAT will continue to evolve and provide utility for evaluating land purchase/acquisition for achieving maximum return on investment in terms of overall contribution to a country's conservation goals. In addition to questions that may be asked about the best remaining core habitat or covering a comprehensive representation of biodiversity, the ultimate question conservation planners want answered is "Where do I get the best ecological return for my conservation dollar?" This question has driven the design of a systematic, logical, and repeatable toolkit that helps planners evaluate activities or events that may be threatening habitat health, identify a comprehensive representation of biodiversity for protection, and configure an optimal portfolio solution for meeting habitat conservation goals. PAT consists of three conservation modules which operate within Environmental Systems Research Institute's (ESRI) ArcGIS 9.3 Geographic Information System (GIS) software:

1) Environmental Risk Surface (ERS)
2) Relative Biodiversity Index (RBI)
3) Marxan Tools

ArcGIS 9.3 software was chosen as the application for the tool since ESRI is a strong supporter of TNC's mission and provides a grant agreement in which conservation partners can freely obtain their software for conservation-related work. Each of the three modules was developed using Visual Basic.NET/ArcObjects. PAT operates on three basic input data layers including a) habitats/species; b) risks elements to habitats/species; and c) protected areas. As this is an iterative process, users are encouraged to continue to refine habitat/species data, goals, and risk elements in order to reassess ecological gaps over time. These tools permit countries to continue the refinement process in a systematic and repeatable manner. PAT will continue to be developed based on user's needs and refined for investigations into a) habitat/species vulnerability to risk elements such as land cover change, fragmentation, and changes to PA boundaries or management practices; b) distributions of habitats/species in the landscape to ensure that there are sufficient occurrences across their environmental range within PAs; and c) overall level of protected status to site irreplaceability as modeled by Marxan.

Mandatory Requirements Needed to Run the Protected Area Tools

1. The user needs to have **ArcGIS 9.3 Desktop** installed.
   - Earlier versions of the tool (e.g. ArcGIS 9.1 and 9.2) can be downloaded from *http://www.gispatools.org* (You can check what version and level of ArcGIS you are running by going to *Help* > *About ArcMap*)
   - An **ArcView Level** is required to run the ERS and RBI modules
   - An **ArcInfo Level** is required to run the Marxan Tools - you also need to have **ArcInfo Workstation** installed. Make sure the **Coverage Tools Toolbox** is loaded (this is the default setting during installation). This toolbox is usually found in C:\arcgis\arcexe9x\Toolboxes (If the user has installed it to the default location)
   - To ensure full functionality of these tools, please install the latest ArcGIS Service Packs from www.esri.com.

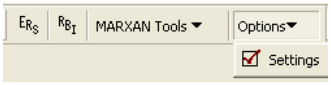
2. Version 3.0 requires that Microsoft .NET Framework 2.0 be installed on your computer. If you do not have Microsoft .NET Framework 2.0 installed, you should follow these steps:
   a) Download the Microsoft .NET Framework 2.0 from http://www.microsoft.com/downloads/details.aspx?familyid=0856eacb-4362-4b0d-8edd-aab15c5e04f5&displaylang=en
   b) Install ArcGIS 9.3 with .Net support (This only shows up after you do step a). If you have already installed ArcGIS 9.3, input the DVD setup disk in your DVD drive and go to Start > Control Panel > Add or Remove Programs. Scroll down to ArcGIS Desktop and click on the Change button. Once the change Menu appears, click on Modify and then install the .NET Framework Support. This option is only available once you install the Microsoft .NET Framework 2.0 in step *a*. Once you have added this support function, you are now ready to install the Protected Area Tools.

3. The user must have installed the **Spatial Analyst** Extension and it should be **turned on.**
   - Check to see if it is available and turned on under *Tools* > *Extensions*)

4. Make sure you your input features are topologically correct by running **geometry checked/repaired** for each input feature. These tools can be found in ArcToolbox under Data Management Tools > Features > Check and Repair Geometry. If problems persist after repairing geometry, converting shapefiles into coverage format, then back to shapefile format will often solve the problem. PAT will process feature classes from geodatabases and shapefiles. Unlike the other modules, the ERS module works with grids and other raster formats supported by ArcGIS.

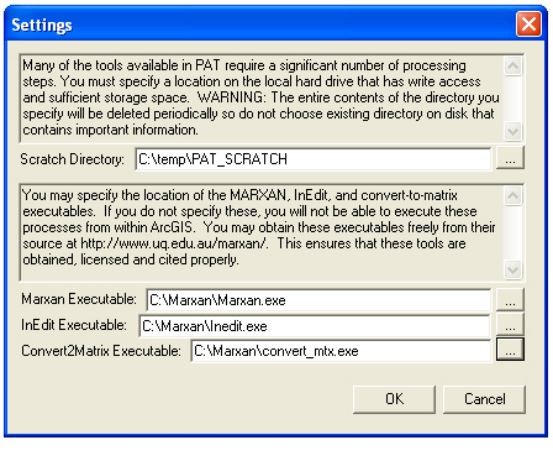
5. All input features must have the **projection defined** and all input features must be in the **same projection.** Since ArcGIS does projections on-the-fly, layers may appear in the same projection, but in reality they are not. You can check the defined projection information by right-clicking on the data layer and going to Properties > Source (Data Source). Remember that you will not be able to calculate correct areas or lengths if you files are in Geographic projection (e.g. decimal degrees). You need to use a projection that uses units such as meters or feet (e.g. UTM, State Plane).

6. For optimal performance, your machine should have at least a 1.0 GHz processor or higher and have at least 1GB of memory/RAM. It is also recommended you have at least 5-10 GB of open hard drive space. It is recommended to operate this program using **Windows XP** as problems may be encountered using Windows 2000 or Vista.

7. 

Many of the tools available in PAT require a significant number of processing steps. You must specify a location on the local hard drive that has write access and sufficient storage space. Please use the **Options > Settings** drop-down menu to specify the location of the scratch folder on your machine (e.g. "C:\temp"). This directory will serve as your PAT_SCRATCH folder were temporary files will be written. WARNING: The entire contents of the directory that you specify will be periodically deleted, so **do not store other vital information in this directory or choose an existing directory on the disk that contains important information**. Remember that depending on the size and level of detail requested by the user, these programs may require large amounts of free disk space to operate. Be sure to check how much space you have on your operating hard drive - **the program will crash if you don't have enough space**. This can especially happen when executing the ERS module, which can create very large, multiple output grids.

8. You may also specify the location of the Marxan, InEdit, and Convert-to-Matrix executables using the **Options > Settings** drop-down menu. If you do not specify where these files reside on your computer, you will not be able to execute these processes within ArcGIS. You may obtain these executables freely from their source at http://www.uq.edu.au/marxan This ensures that these tools are obtained, licensed, and cited properly.


Settings
Many of the tools available in PAT require a significant number of processing steps. You must specify a location on the local hard drive that has write access and sufficient storage space. WARNING: The entire contents of the directory you specify will be deleted periodically so do not choose existing directory on disk that contains important information.
Scratch Directory:
C:\temp\PAT_SCRATCH
You may specify the location of the MARXAN, InEdit, and convert-to-matrix executables. If you do not specify these, you will not be able to execute these processes from within ArcGIS. You may obtain these executables freely from their source at http://www.uq.edu.au/marxan/. This ensures that these tools are obtained, licensed and cited properly.
Marxan Executable:
C:\Marxan\Marxan.exe
InEdit Executable:
C:\Marxan\Inedit.exe
Convert2Matrix Executable:
C:\Marxan\convert_mtx.exe
OK
Cancel

Installing the Protected Area Tools (PAT) v.2

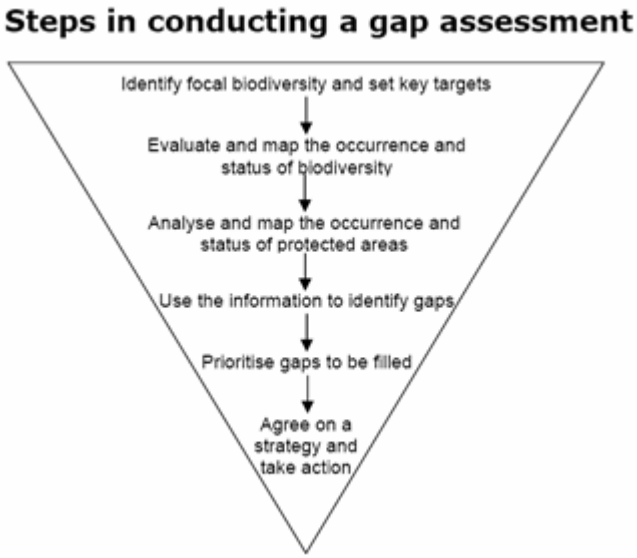
MAKE SURE YOU HAVE Microsoft .NET Framework 2.0 installed AND the .NET Support enabled in ArcGIS prior to installing PAT. Failure to do so will prevent the tools from working properly. This functionality is NOT installed in the default setup of ArcGIS. For more information, see number 2 above.

Download the zip file "*patv2.zip*" from *http://www.gispatools.org*. This file contains the "setup.exe" executable file that installs the tool in your ArcGIS bin directory and makes the tool available in your View > Toolbars menu. Remember that version 3.0 of the program only works with ArcGIS 9.3. For those using ArcGIS 9.1 or 9.2, an older and less functional (version 1.0) of the tool is available at *http://www.gispatools.org*. Please note that versions 1.0 and 2.0 have less functionality and are no longer supported.

Once you have downloaded and unzipped the *patv2.zip* file, double click on the setup.exe file and the wizard will walk you through the installation process. When the installation is complete, launch ArcMap and turn on the extension by going to View > Toolbars > TNC Protected Area Tools. You will see the following icon appear in your ArcMap window:

You can drag this icon, by clicking on the left sidebar and positioning anywhere in your ArcMap toolbar. This new toolbar gives you access to all the functionality in the three modules of the Protected Area Tools: ERS (Environmental Risk Surface), RBI (Relative Biodiversity Index), and the Marxan Tools.

The Protected Area GAP Assessment Process


Steps in conducting a gap assessment
Identify focal biodiversity and set key targets
Evaluate and map the occurrence and status of biodiversity
Analyse and map the occurrence and status of protected areas
Use the information to identify gaps
Prioritise gaps to be filled
Agree on a strategy and take action

Identifying gaps in protected areas is one step in the demanding and often daunting process of creating country-level ecologically-representative protected area networks. Although there are a variety of ways and methods that protected area gap assessments can be conducted, several principles and guiding methods are outlined in Dudley and Parrish (2006). The tools presented in this tutorial are aimed at supporting several steps of this process, but the user must always remember that the model output is only as good as the quality of the input data. The rule of "garbage in, garbage out" cannot be emphasized enough. It is recommended that 80-90% of the total gap process work should involve careful thought and consideration with in-country experts

for focal habitats/species delineation, conservation goal setting, and defining/ranking of risk elements that threaten the health of targeted habitats or key species.

The guiding principles for conducting a protected area gap analysis include (Dudley and Parrish, 2004):

1. *Ensure full representation* across biological scales (species and ecosystems) and biological realms (terrestrial, freshwater, and marine).
2. *Aim for redundancy* of examples of species and ecosystems within a protected area network to capture genetic variation and protect against unexpected losses.
3. *Design for resilience* to ensure protected area systems to withstand stresses and changes, such as climate change.
4. Consider *representation gaps*, *ecological gaps* and *management gaps* in the analysis. Representation gaps refer to species, ecosystems and ecological processes that are missed entirely by the protected area system; Ecological gaps relate to biodiversity that exists within protected areas but with insufficient quality or quantity to provide long term protection; while management gaps refer to situations where protected areas exist but are failing to provide adequate protection either because they have the wrong management objectives or because they are managed poorly.
5. *Employ a participatory approach*, collaborating with key experts and stakeholders in making decisions about protected areas.
6. *Make protected areas system design an iterative process* in which the gap analysis is documented, reviewed and improved as knowledge grows and environmental conditions change.

Assemble Team of Experts
Identify and Engage Key Partners/ Stakeholders
Identify Conservation Targets
Identify Threats to Conservation Targets
Identify PA Boundaries and Management Practices
Gather and Create Geospatial Data
Fill Data Gaps Based on Expert Review
Screen Targets and Set Goals
Conduct PA GAP Analysis
Expert Review and Modification of Analysis if Needed
Organize Data and Final Product of Results
Reassess Once New Data Becomes Available
PROTECTED AREA
GAP ASSESSMENT
PROCESS

The majority of the work that goes into a protected area gap assessment involves the spatial delineation and critical evaluation of habitats/species, protected areas, and risks to focal habitats. The Protected Area Tools are ready to be used only after users have obtained the highest quality data available, conducted an ecological inventory and assessment of these data layers through expert review, and carefully considered all model scenario settings. Three modules presented in this tutorial will guide users through the process of

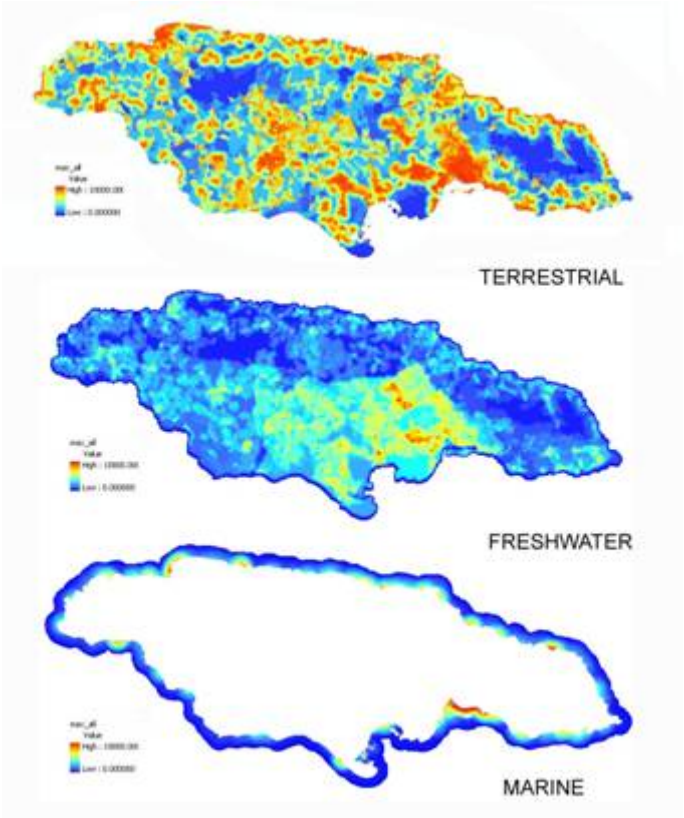
1) Developing a customized *Environmental Risk Surface (ERS)* based on mapped risk elements (*i.e.* socio-economic activities) that have been identified through expert review as having negative impacts on the health of targeted habitats, species or ecological systems;
2) Calculating a landscape's *Relative Biodiversity Index (RBI)*, which measures relative rareness or uniqueness, measured in terms of biodiversity feature abundance in comparison to the overall study area. Individual scores for each biodiversity occurrence can be used as a stand alone assessment for each planning unit or subsets of units (*e.g.* hexagons, watersheds);
3) Creating input files and viewing model results for Marxan, powerful software which provides users an easy way to manipulate input parameters and test/review various conservation scenarios in order to achieve an optimal configuration of protected areas that meet user defined conservation goals.

This tutorial is not meant in any way to cover all facets of protected area gap assessments. For a complete review of this process and for additional recommendations and guidelines on conducting a protected area gap assessment, please refer to Dudley and Parrish (2004).

MODULE 1: Environmental Risk Surface (ERS)

One of the primary goals of many conservation portfolio selection approaches is to create a functional landscape or network of sites that support all elements of biodiversity. One of the key aspects of this approach is minimizing environmental risk to critical habitats and key species. Although not all human activities can be considered risks to biodiversity, direct or indirect human impacts are ultimately responsible for most alterations of the ecological processes that sustain biodiversity. In addition, there are other events that pose significant risk to habitats and species (*e.g.* hurricane, volcanic activity, climate change). Consequently, understanding the spatial relationship between these risk elements and ecological health within focal conservation sites provides valuable insight into conservation management. However, assessing and predicting risks to habitats represents one of the most challenging dimensions of conservation planning due to the unpredictability, wide variation, and lack of existing information regarding the functional relationships of ecological processes to impacts in terrestrial, freshwater, and marine realms. Standardized and widely accepted methods for quantifying the relative degree of impacts and utilizing risk measures to prioritize areas for conservation is in its infancy and will surely be more fully developed in the coming years.

An Environmental Risk Surface (ERS) is a modeled surface developed using mapped risk elements (*e.g.* socioeconomic information) to explore the overlap between these risk elements and biodiversity features. A *risk element* can be defined as anything identified by experts as having a negative influence on the health of a critical habitat or key species. One can also refer to risk elements as a *threat*. The ERS measures cumulative levels of risk impacts across the landscape and can be used to focus conservation site selection by steering habitat selection away from high-risk areas where the abatement of pressures on biodiversity seems less likely. The composite surfaces or disaggregated individual surfaces can be used to get a better idea of the specific environmental risks on the landscape that may be degrading the viability of certain conservation habitats/species (*i.e.* targets). ERS model output can also be used to assist in the screening of habitats and creating cost surfaces for Marxan runs.

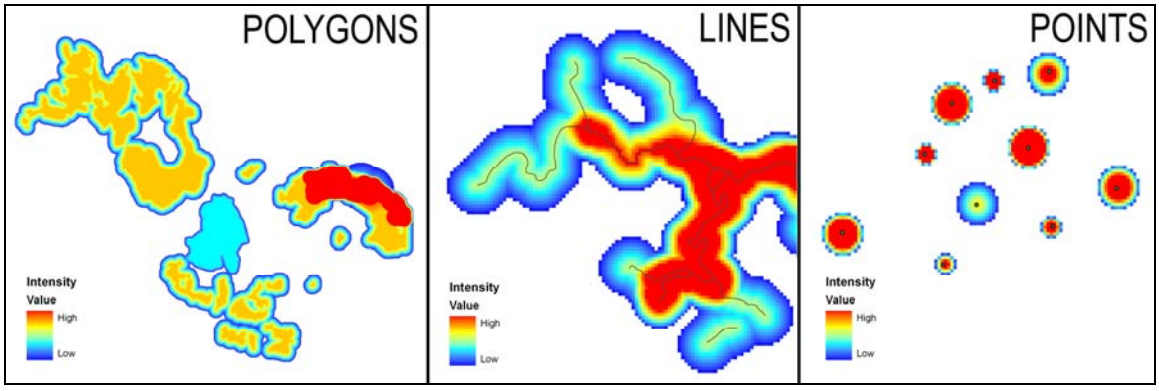


Examples of customized Environmental Risk Surfaces (ERS) models for Jamaica's terrestrial, freshwater, and marine realms. Red areas indicate higher risk to habitats based on the aggregation of intensities of underlying risk elements (McPherson et al, 2008)

Creating an ERS model first requires assembling a suite of the best available GIS

data to spatially represent the specific risk elements (*e.g.* human activities) most likely to impact critical habitats or key species. This means all risk element features must be spatially mapped on the landscape with precise location and boundaries (when possible) using expert opinion or obtaining (or creating through on-screen digitization) the most accurate GIS layers available. Human activities are often the most common risk element features used to create ERS models. Activities such as agriculture, urbanization, tourism zones and hotels, roads, industry, and population density are examples of risk elements that can be used in the creation of an ERS. These models can be developed specifically for terrestrial, freshwater, and marine realms, based on available input data and expert assessments for each risk element.

Once all input data are gathered, experts must then review and rank each risk element to the degree that it is a threat to the habitat/species in question. This is done by assigning three variables to each point, line, or polygon feature that represents the corresponding risk element. The three variables assigned are *intensity value*, *influence distance, and distance decay function*. These values can be derived through expert evaluation of the extent, severity, and reversibility of each risk element in relation to the conservation target(s). In addition to socio-economic data, natural event data for events such as hurricanes, volcanic activity, and climate change may also be used in the creation of an ERS, although it may be difficult to rank and assign these values due to the extreme unpredictability and complexity of these events. The figure below shows examples of polygon, line, and point risk elements that represent modeled risk surfaces with varying intensity values and influence distances. The dark red areas represent higher combined risk and the lighter blue areas, lower risk as modeled by the mapped risk element features. Examples of how to assign intensity values, influence distances, and different distance decay functions are explained in the next sections.



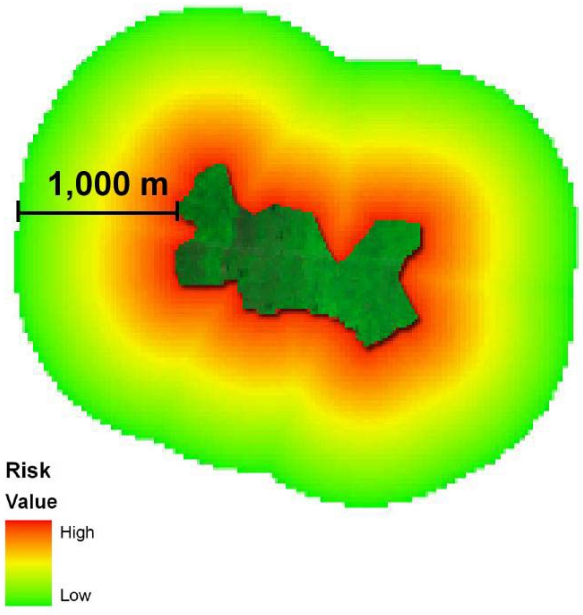
Examples of Environmental Risk Surfaces (ERS) derived from polygon, line, and point risk element features. Each risk element feature has an intensity, influence distance, and distance decay function assigned based on the potential threat to biodiversity health. Red represents higher risk values which decreases linearly as the distance is increased away from the combined risk element features (*i.e.* blue represents lower risk).

1.0.1 Intensity Value

Having identified the risk elements, experts then rank each element in relation to each other and assign an intensity value in terms of degree of risk to the focal habitat/species. For example, certain types on mining, such as bauxite, may have more damaging effect on freshwater systems and thus assigned a higher intensity value, compared to limestone excavation which may have less damaging effect. Intensity values and influence distances for each risk element can be defined using information from existing literature and/or expert opinion regarding the impacts of these activities or features on ecosystems. Separate intensity values can be assigned relative to their impact on terrestrial, freshwater or marine biodiversity. The intensities should be normalized based on a relative scale (*e.g.* 0.0 – 1.0 or 0 - 100) so that they are comparable across all risk elements and classes (*i.e.* different types of roads). The normalized intensity scores are not a representation of the absolute measure of the impact of activities on biodiversity. Rather, these normalized values should be a relative degree to which biodiversity is more likely to survive in one place over another based on the presence of a given activity in comparison to another activity. Examples of intensity values used in a conservation assessment for Jamaica can be found in McPherson et al, 2008.

1.0.2 Influence Distance

After the intensity values have been assigned, the next step is to determine the influence distance of each risk element. The influence distance is the spatial extent or footprint of the activity and represents the maximum distance the feature has a negative impact on a biodiversity. In other words, as the distance of the buffer increases away from the center (point, line or polygon) where the activity is taking place, the intensity values of the cells within the buffers diminish progressively (distance decay) and the risk to the habitat is lessened. Some features may have high risk activities that extend a long distance from the center, while others have a lower risk with more constricted boundaries. It is best to set up a risk element table that indicates the intensity and influence distance of each risk element. This will be explained in more detail in the next section.

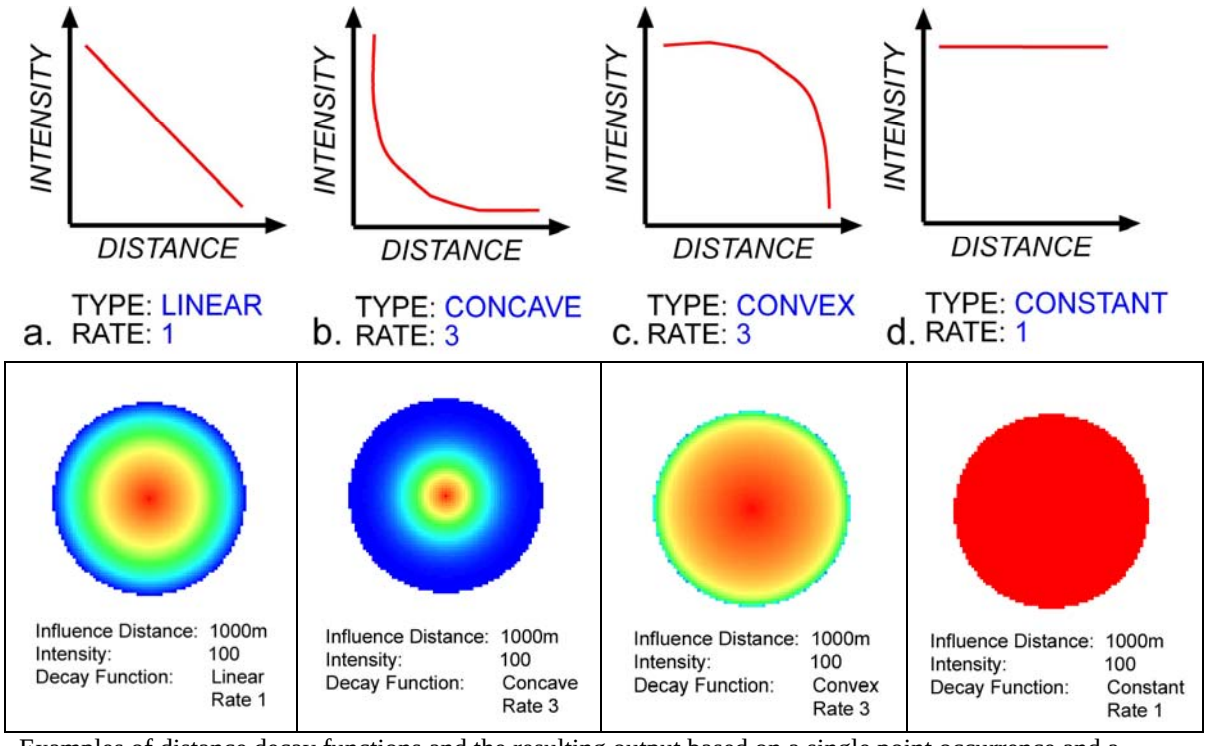


1.0.3 Distance Decay Types

Ecologists have documented that depending on the ecosystem, some risk elements may have a different impact than on other risk elements (Ervin and Parish 2006, Theobald

2003, Araújo et al. 2002). In some cases, there may be a direct linear relationship between a risk element and an ecosystem's response to the threat. Often times, the greater the expression of a risk element, the larger that threat may have upon an ecosystem. However, this is not always the case because some ecosystems have different characteristics that govern the way they might respond to a given threat. For example, fire in certain ecosystems can be a disaster and immediately destroy or highly degrade an ecosystem, such as rainforests in the Amazon. But in grasslands such as in the Orinoco Basin, it is necessary to have fire in certain intensities and frequencies. To little fire is bad for that ecosystem, and too much fire is equally bad. There is a non-linear response relationship between a grassland and fire that has to be identified and modeled (TNC, 2006).

Prior to modeling an ERS, experts must identify how risk in each element decays over distance, or if there is a decay factor at all. PAT provides four decay types for expressing the function of distance decay. These include a) linear; b) concave; c) convex; and d) constant and are represented in the figure below. The default function is a *linear* decay type, where the rate of intensity decay is constant until the maximum distance is reached and the intensity becomes zero. A *concave* decay has an initial rapid decrease in intensity while a *convex* has the opposite effect - a gradual intensity decay followed by a steep decay as maximum distance is reached. A *constant* decay has no change in intensity value until the maximum distance is reached, ending in an abrupt zero. Users should consult with ecological experts who have reviewed each risk element and defined the types of decays that are appropriate to apply based on the ecosystem in question.



Examples of distance decay functions and the resulting output based on a single point occurrence and a 1000m influence distance. Red and orange hues indicate higher intensity values while green and blue hues represent lower intensity values.

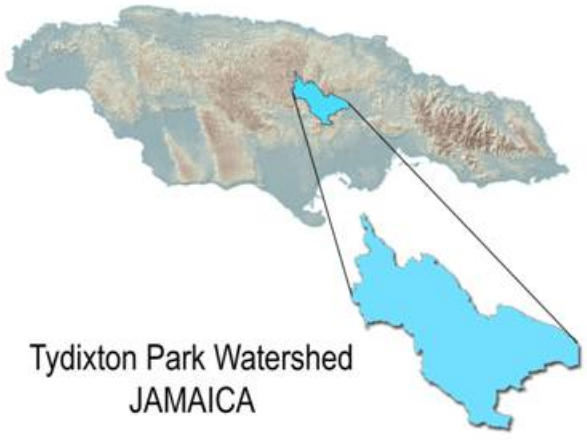
EXERCISE 1 Building an Environmental Risk Surface

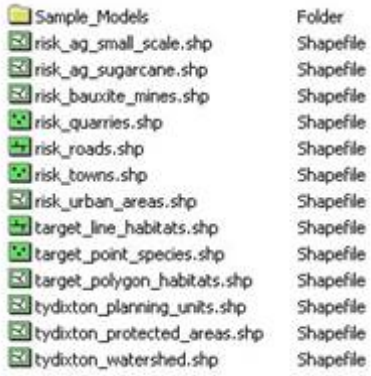
REMEMBER

- Every intensity field must be a whole (integer) number, **NOT floating point with decimals**. The program creates an output grid for each unique intensity value, thus it could take a LONG time run if you have hundreds of unique floating point combinations. If you have floating point intensity values, please rescale to integers with an appropriate number of unique values (depending of computer hard drive free space and processing power)
- All influence distances under the user-specified cell size will automatically default to the user-specified cell size. Remember to enter an output cell size that appropriately fits your defined influence distances.
- Check how much available hard drive space you have on your computer - The program will crash if you don't have enough operating space (*e.g.* 2-3 GB of space depending on scale, resolution, and complexity of your input features).
- All risk element features (shapefiles, geodatabases, and grids) must be in the same projection and have the same projection information defined. When using grids, the grid value is used as the intensity value.

Sample Data: Tydixton Park Watershed, Jamaica

The sample spatial data used in these tutorial exercises comes from the Tydixton Park watershed in central Jamaica. The data contains habitat features representing lines (rivers), points (endemic species locations), and polygons (forest types). In addition to habitats/species, there are shapefiles representing various socio-economic activities, or habitat risk elements. These include agriculture, mining, roads, and urban areas. The other files are the planning units (hexagons), protected areas, and the boundaries of the Tydixton Park watershed. All of these files will be used to conduct a preliminary protected area gap assessment based on the results of the three exercises. As with any input file that is used in PAT, each data file must have any geometry errors corrected and projection information defined. These sample datasets have been projected to Lambert Conformal Conic using the Jamaican Datum 2001 (JAD2001).





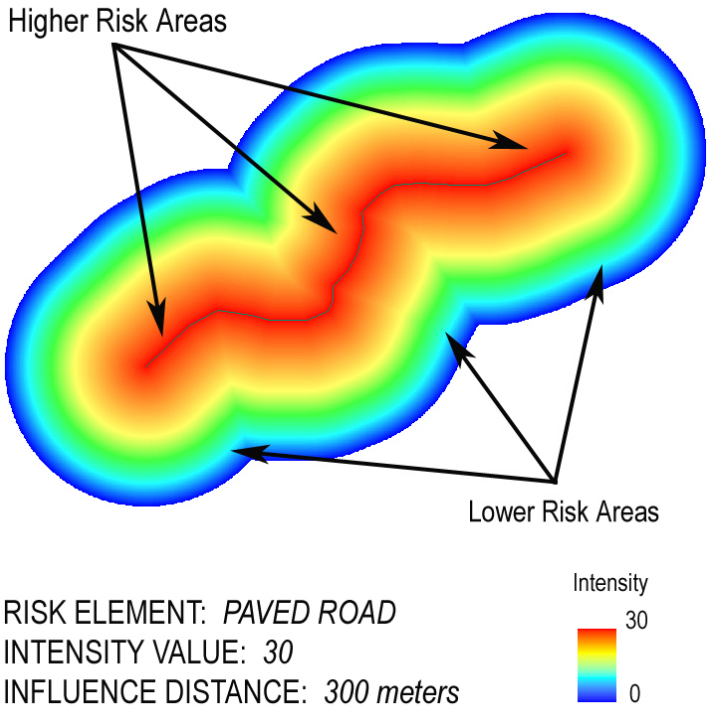
1.1 Setting up the ERS Table

Successful conservation planning involves risk management. ERS models are about categorizing and ranking potential risks to habitats or features on concern. The first step in creating an Environmental Risk Surface is to create an intensity value and influence distance table for all identified risk elements and associated classes. These values can be customized specifically for the creation of ERS models for terrestrial, freshwater, or marine realms. Remember that each risk element has to be thought of in terms of its impact on the habitats or species in the focal realm (*i.e.* terrestrial, freshwater, and marine). A sample Excel spreadsheet risk matrix table is provided with the tutorial data, named "*Sample_Intensity_Table.xls*." Open this risk matrix table and you will see six columns that represent the *risk element*, the associated *class* for each risk element, followed by the terrestrial and freshwater intensity value and influence distance columns. For instance, the file "*risk_quarries.shp*" contains two classes, "*River Aggregate*" and "*Limestone*". According to the experts, mining for limestone is more damaging to terrestrial systems than river aggregate mining. Consequently, limestone mining is assigned an intensity value of 40, compared to river aggregate, which is assigned 30. Likewise, the file "*risk_roads.shp*" has four road class types and corresponding intensity values have been assigned based on expert opinion of the degree of potential risk to critical habitats.

RISK ELEMENT	CLASS	TERRESTRIAL INTENSITY	TERRESTRIAL INFLUENCE DISTANCE	FRESHWATER INTENSITY	FRESHWATER INFLUENCE DISTANCE
risk_ag_small_scale.shp		25	500	25	3000
risk_ag_sugarcane.shp		11	1000	11	5000
risk_bauxite_mines.shp		50	1000	60	8000
risk_quarries.shp	River Aggregate	30	500	10	1000
	Limestone	40	500	20	1000
risk_roads.shp	TRACK	5	30	5	30
	OTHER	10	30	10	30
	CLASS B	15	60	15	60
	CLASS A	30	60	60	120
risk_urban_areas.shp		95	500	95	1000

It is important to remember that when assigning intensity values and influence distances, you are using a relative scale, not a representation of the absolute measure of the impact of activities on biodiversity. These values should be a relative degree to which biodiversity is more likely to survive in one place over another based on the presence of a given activity in comparison to the other risk elements. For this exercise, we use a scale of 0-100, with 100 being the highest degree of risk to a habitat. It is up to the user to define the scale, but the user should use whole numbers (integers) for computational reasons. The influence distance should be in whatever units your input data is in (*i.e.* projection information). In the sample data, the units are in meters. As with intensity

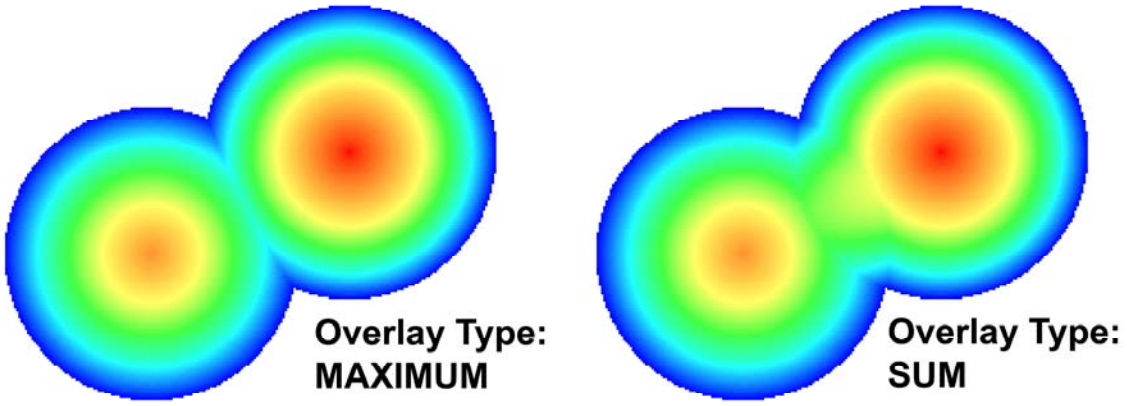
values, expert opinion should be used in assigning a maximum influence distance that the risk element may have on a particular habitat or realm. The program calculates intensity decay from the edge of each risk element using the user-specified influence distance and distance decay function. For example, in the file called "*risk_ag_sugarcane.shp*" there is a terrestrial influence distance of 1000 meters, or 1 kilometer. If the user chooses to use a linear decay function (default), the intensity value of 11 will be assigned within close proximity of the boundaries of each sugarcane polygon, but as you move away from the polygon, the intensity becomes linearly smaller and smaller until you reach a distance of 1000 meters, where the intensity is reduced to zero (0). In the figure below, there is an example of an ERS model output for a paved road with an intensity value of 30 and an influence distance of 300 meters. Notice the linear decay function that is applied like a buffer completely around the risk element in all directions. Additional guidance on setting up intensity values and influence distances for a variety of socioeconomic activities can be found in (McPherson et al., 2008).



An example of ERS model output for a paved road risk element with an intensity value of 30 and an influence distance of 300m. Notice the linear decay function that is applied like a buffer completely around the risk element in all directions.

1.1.1 Overlay Function

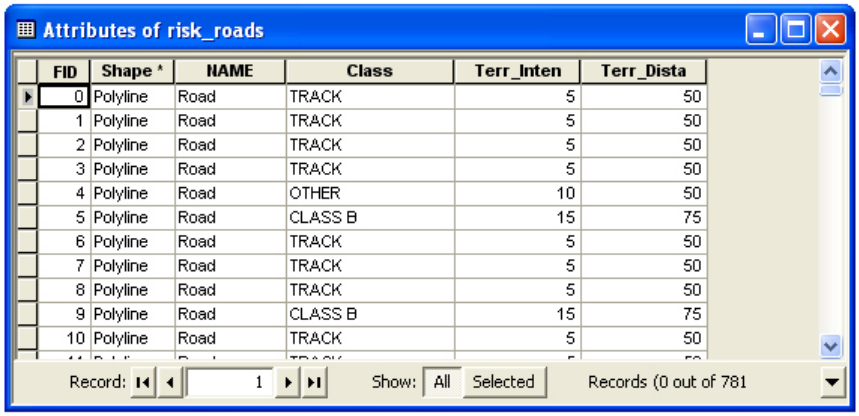
Another item to consider is how to resolve the overlaying areas of influence distances between neighboring risk element features. The ERS tool allows users to specify how risk elements will be combined based on map algebra statistical functions (e.g. mean, majority, maximum, median, minimum, minority, range, standard deviation, and variety). These are further explained in the next section, but if influence distance overlay between risk element features is anticipated, it is important to think about the best way to resolve this type of aggregation prior to executing the tool. The default function for influence distance overlay is SUM, calculating the sum for all layer values. Most often, either SUM or MAX are the most typical functions used. Examples of each are given in the figure below.



Examples of the most commonly utilized overlay function types (Maximum and Sum) for resolving overlaying areas of influence distance between neighboring risk element features. Notice the differences in how each function handles the computation on the intensity values within the overlay areas.

1.2 Adding and Assigning the Intensity and Influence Distance to Each Risk Element Feature

Prior to executing the ERS model, all risk elements must first have intensity and influence distance values assigned. As explained in the previous section, creating an ERS risk matrix is helpful when comparing intensity and influence distance values and ranking risk potential between risk elements. The user has the option of entering these values manually using the ERS tool, or defining these values in a field within the corresponding shapefile or geodatabase. For large and complex datasets, it is always easier to define the values using attribute fields. The intensity and influence distance fields should be defined as integer types and given a name that the user can recognize. If the risk element has a class definition listed in the attribute table, it is important to assign the correct values to each corresponding class type. The risk matrix can help you when it is time to calculate these fields. Remember that a raster dataset will be created for each unique combination of risk element class intensity, and influence distance values. If you have a large number of classes, intensity, and influence distance values, the computation could take a *very long time* to complete. It is recommended that the user first test the model using one or two risk element files in order to get an idea of optimal output cell size, run time required, and hard disk space requirements. Prior to launching the ERS module, you must also think about how to resolve overlay areas within the influence distance and make sure that each risk element is in the same projection with the projection information properly defined. Risk element overlay functions should be determined by experts who understand the how these elements interact spatially.

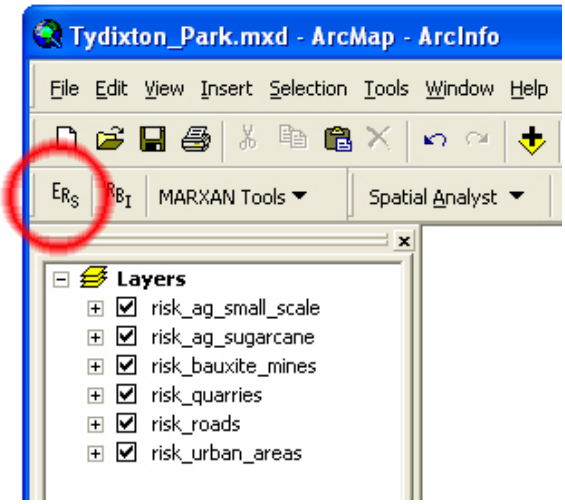

Attributes of risk_roads
FID
Shape *
NAME
Class
Terr_Inten
Terr_Dista
0 Polyline Road TRACK 5 50
1 Polyline Road TRACK 5 50
2 Polyline Road TRACK 5 50
3 Polyline Road TRACK 5 50
4 Polyline Road OTHER 10 50
5 Polyline Road CLASS B 15 75
6 Polyline Road TRACK 5 50
7 Polyline Road TRACK 5 50
8 Polyline Road TRACK 5 50
9 Polyline Road CLASS B 15 75
10 Polyline Road TRACK 5 50
Record: 1
Show: All Selected
Records (0 out of 781

An example of the attribute table for *risk_roads.shp* indicating class (road type) and corresponding intensity and influence distance values.

1.3 Executing the ERS Module

Once risk element features have all been assigned intensity and influence distance values and the layers are in the same projection, the user is ready to execute the ERS module. Prior to clicking on the ERS button in the PAT toolbar, make sure all the risk element

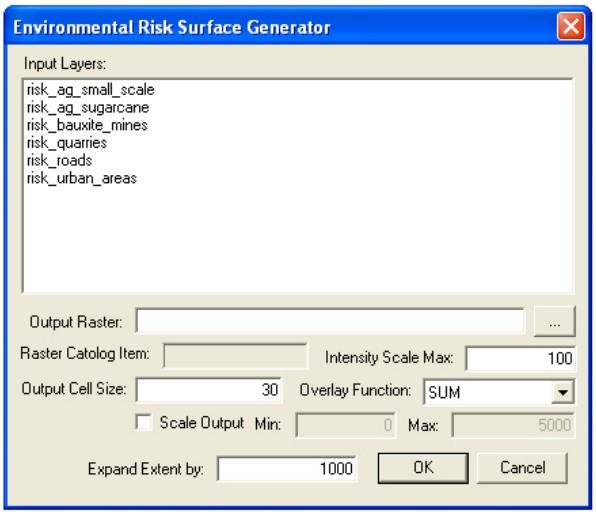
features have been added to the map view and that the Spatial Analyst Extension is turned on by going to *Tools* > *Extensions*. Remember that shapefiles, geodatabases, and grids can all be used in the ERS module. When using grids, the grid value is used as the intensity value. For this exercise, add the risk element features that are included in the figure below, to the map view. These features are part of the Tydixton Park tutorial sample dataset and are in the same projection (JAD2001) with the projection information defined (*i.e.* prj file).



It is important to note that since ArcGIS calculates projection on the fly, layers can appear to be in the same projection, but may not actually be in the same projection. This is important to understand because when executing an ERS model, all layers are ultimately turned into grids before a series of map calculations are made. Since Spatial Analyst does not do projection on the fly, having the same coordinate system for all input layers is critical for success in these map calculations.

1.3.1 Launch the ERS Module and Specify Input Layers and Output Parameters

You are now ready to launch the ERS module by clicking on the ERS button in the toolbar. This will bring up the ERS interface that looks like this:



a. Select the risk layers that you want to include in the ERS analysis by clicking on them in the ERS interface window. Once selected, they appear highlighted and will be used in the model. Click to unselect.

b. Specify an **Output Raster**. This should be a valid raster name. You can save your output GRID, IMG, personal geodatabase, file geodatabase, ArcSDE geodatabase, or non georeferenced formats (BMP, GIF, JPEG, JPEG 2000, PNG, TIFF). When storing a raster dataset in a geodatabase, no file extension should be added to the name of the raster dataset. When storing the raster dataset in a file format, you need to specify the file extension: .bmp for BMP, .gif for GIF, .img for an ERDAS IMAGINE file, .jpg for JPEG, .jp2 for JPEG 2000, .png for PNG, .tif for TIFF, or no extension for GRID. The output grid should not be written to a directory that has a space in the name or in the directory path.
   - The output grid name cannot be over 13 characters long.
   - The output grid name cannot contain spaces. Underscores are permitted.
   - The output grid name cannot contain special characters; for example, #, @, %

   The output grid name cannot begin with numbers. You may also specify a raster catalog location to save your output to. If you specify a raster catalog output, you must specify the name for the raster catalog item in the "Raster Catalog Item" box described below. This box will be grayed out unless you have specified a raster catalog output. When specifying a raster catalog location for output, your final result will not be added to the map. You will need to manually load the data into ArcMap.

c. If you have specified your output as Raster Catalog specify a name for the **Raster Catalog Item** that will be created.

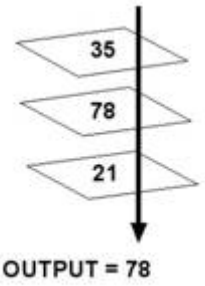
d. Enter the **Intensity Scale Max** field. This represents the theoretical maximum value for intensity. It represents the maximum value in the scale of all your intensity values. In other words, if you have established your intensity values on a scale from 0.0 – 1.0 enter a value of 1 in the Intensity Scale Max field. The tool does not support using a value other than 0 as the minimum scale value. You may or may not actually have any intensity values assigned to the value you enter here. For the tutorial data, we will use a scale of 0-100, so enter the value of 100 as the maximum intensity value.

e. Enter the **Output Cell Size.** This is the desired cell size of the final output grid. The default value is 30m. Remember that the smaller the cell size, the longer the computation time (exponentially) and the more hard disk space required. Run time also depends on the scale you are working at and the number of unique intensity values that you have defined. It is a good idea to experiment with cell sizes to get an idea of run time and space requirements. All input distances need to be specified in the same units as the input data (*i.e.* all inputs should use the same units). Also, if you have defined an influence distance less than the cell size,

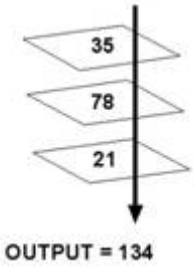
the output will automatically default to the user-defined cell size. For example, if roads have an influence distance of 15m and a cell size of 30m is used, the cell size will override the system and the road influence distance will be 30m. For the tutorial example, we will use a cell size of 30 which corresponds with the scale of the input data. You can experiment using different cell sizes to get an idea of how the resolution and file size correspondingly change.

f. Specify the **Overlay Function.** This is where to specify the map algebra function that you want used when combining all of the output risk element groups. A risk element group means all of the risk elements created by a single input layer (e.g. roads, towns, quarries). Later you will be given the option of specifying the overlay function *within* a risk element group (e.g. types of roads). For the tutorial example, we will use the default SUM function, which is the more commonly used function, since it aggregates all risk element intensity values.

MAXIMUM – Takes the maximum grid value for each cell in all computed intensity layers. For example, if cell values between layers are 35, 78, and 21, the final cell number would be 78.



SUM - Takes the sum of the grid values for each cell in all computed intensity layers. For example, if cell values between layers are 35, 78, and 21, the final cell number would be 134.



Other less used overlay functions available for use include:

MEAN: Takes the arithmetic average of the values between input rasters on a cell-by-cell basis. The mean provides a measure of the center of the distribution of the values.

MINIMUM - Takes the minimum grid value for each cell in all computed intensity layers. For example, if cell values between layers are 35, 78, and 21, the final cell number would be 21. This is not a common option to choose since a value of 0 (no risk) in any location in any risk element input will produce a value of 0 in that location in the final output grid regardless of any of the other input values. In other words in areas of no overlap between risk elements the risk assigned will be 0.

MAJORITY: Takes the majority value or the value that appears most often between input rasters on a cell-by-cell basis.

MEDIAN: Takes the median value that corresponds to a cumulative proportion of 0.5. If each input cell value was arranged in increasing order, 50 percent of the values would lie below the median, and 50 percent of the values would lie above the median. The median provides another measure of the center of the distribution.

MINORITY: Takes the value that occurs the least often between input rasters on a cell-by-cell basis.

RANGE: Takes the range of values (highest to lowest) between input rasters on a cell-by-cell basis.

STD: Takes the standard deviation of the values between input rasters on a cell-by-cell basis. The standard deviation is the square root of the variance. It describes the spread of the data about the mean in the same units as the original measurements. The smaller the variance and standard deviation, the tighter the cluster of measurements about the mean value.

VARIETY: Takes the number of unique values between input rasters on a cell-by-cell basis.

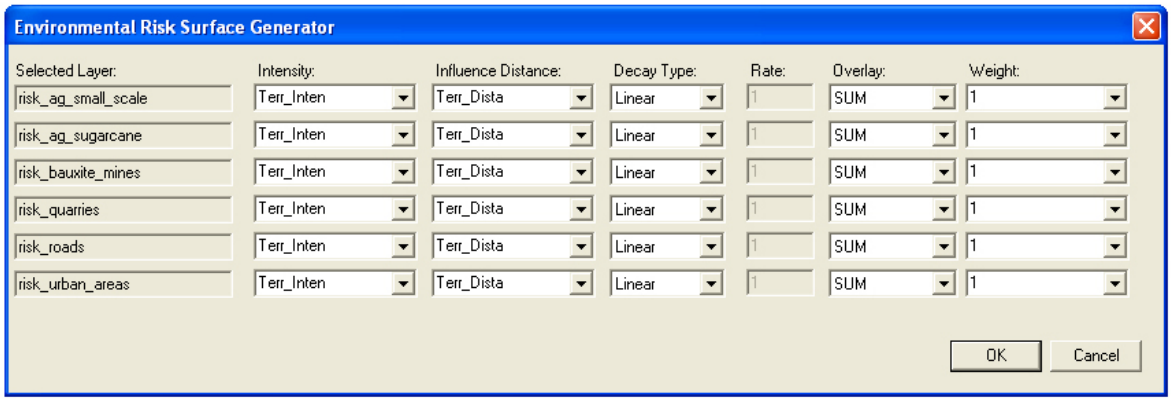
g. You can also use the **Scale Output** feature to rescale the final output model to a specified range. This is useful for defining scale ranges in cost surfaces to be used in Marxan. For example, to keep a tight cost range, use the area of a hexagon as the base and then five times the area as the maximum cost value (*i.e.* 260-1300). For the tutorial example we will not use this option, but it may be useful if you are trying to keep your intensity values within a certain range for comparative purposes.

h. Enter the **Expand Extent By** Value. This value represents the distance that final extent will be extended beyond the maximum spatial extent of all the input layers. This will prevent the output grid intensity values along the edges from being inadvertently cut off. Commonly, this is the maximum influence distance that is used on any one of the input risk elements. For the tutorial example, we will use the default of 1000m. On some occasions it is desirable to enter a greater extent if further processing will be required using datasets outside the current extent.

Once you have all the input parameters entered, click **OK**.

1.3.2 Specify the Intensity, Influence Distance, Decay Type, Overlay, and Weight for each Risk Element

Once you click OK in the previous menu, a new dialog box will appear that allows the user to specify the values for *Intensity*, *Influence Distance*, *Decay Type (Rate)*, *Overlay*, *and Weight* for ***each*** input risk element feature. The name of each selected risk element feature will appear on the left side of the dialog box. The user has the option of typing these values in, or clicking on the down arrow and choosing a field that contains the corresponding values. If the user is going to type in the values, the ERS risk matrix previously prepared should be opened so the correct values for each element can be typed in.



For this exercise, we are going to use the influence distance and intensity values that have been previously assigned in the shapefile attribute fields. Since we are computing an ERS model for a terrestrial example, the fields have been defined as "Terr_Inten" for intensity and "Terr_Dista" for the influence distance. If the user needs to generate a variety of ERS scenarios for modeling risks for an array of species or ecosystems, several different fields could be created that represent the intensity and influence distance values for each corresponding species or ecosystem. As previously explained, the *Decay Type* refers to the spatial function that will be applied to the intensity values as distance away from the risk element increases (i.e. LINEAR, CONCAVE, CONVEX, and CONSTANT functions).

The Overlap function on this screen allows users to choose how to combine individual risk elements contained in each layer. For example, if the user has specified roads with different levels of intensity, the overlay function specified here will be used to calculate how to resolve areas where two roads of different intensities meet (within the same risk element features). Although SUM is the default, there are instances where MAXIMUM may be a better option. You may choose any of the overlay functions previously explained (e.g. mean, majority, maximum, median, minimum, minority, range, standard deviation, and variety). The *Weight* field is used to assign higher weight criteria to individual risk element features. The default is set to 1, meaning all features are weighed equal in the computation of the final risk surface model. The weight value behaves much like the intensity value so caution should be used when altering this value.

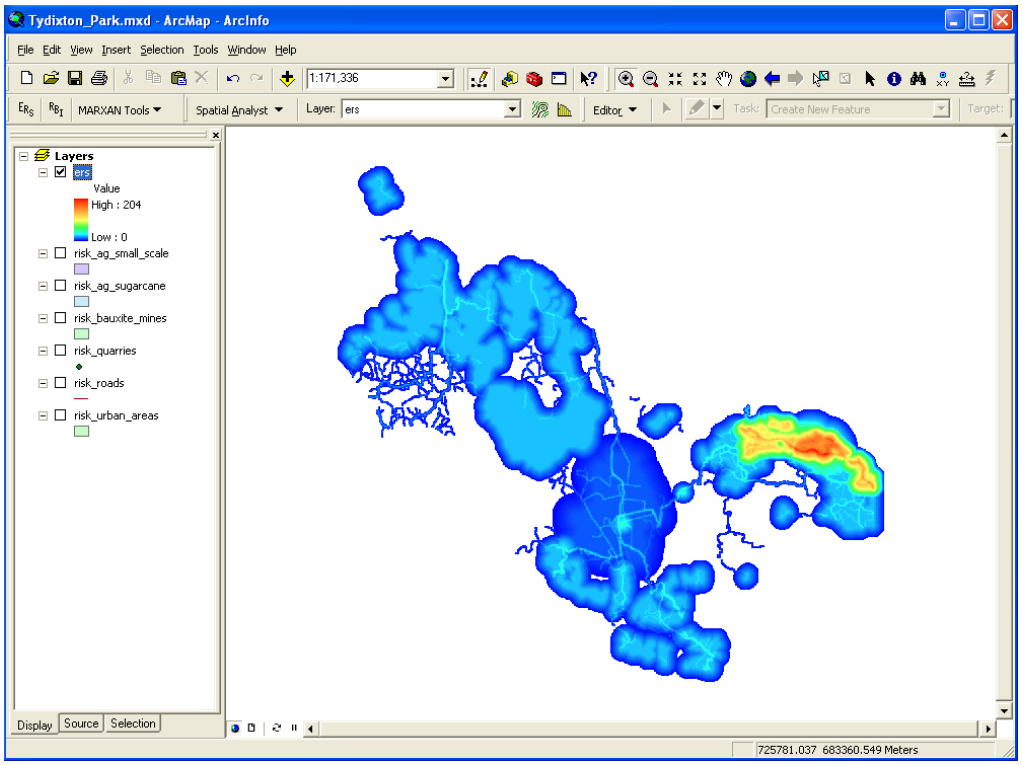
Now select the appropriate intensity, influence distance, decay type, overlay, and weight function values manually or by using the drop down menus. For this exercise, choose the "Terr_Dista" for influence distance and "Terr_Inten" for the intensity values for all input

layers (risk element groups) and leave the default values for the remainder of the input parameters.

If you want to see the commands as they are executed, you need to open your Command Line Window by clicking on the Open/Close Command Line Window button in the ArcGIS menu bar. When ready, click OK to execute the model. You will notice in the lower left-hand corner of ArcMap, the risk element layer number that is being processed. As the processing consumes much of the computer's resources, it is best not to work on the computer or click on ArcMap while the model is executing. Once the model is finished, a text box stating "Processing Complete" will appear and the results will be loaded into the map view.

1.3.3 Viewing the ERS Results and Summarizing Spatial Statistics

When the ERS model has finished processing, the final grid will appear in your map view and have the spatial extent of the boundaries of all specified input layers. The output grid is displayed in the view with the background values set to zero, but the user can modify this and other display parameters using *Layer Properties* > *Symbology*. All of the intermediate processing steps are saved into the specified PAT scratch directory. This directory is deleted every time ArcMap is started.



Zonal Statistics in Spatial Analyst can now be used to summarize the ERS surface statistics by planning unit hexagon or habitat polygon. In doing so, planning units can be assigned a mean cost value for use in Marxan or habitat polygons/lines/points can be ranked by the underlying ERS area-weighted statistics. In order to do this, each planning

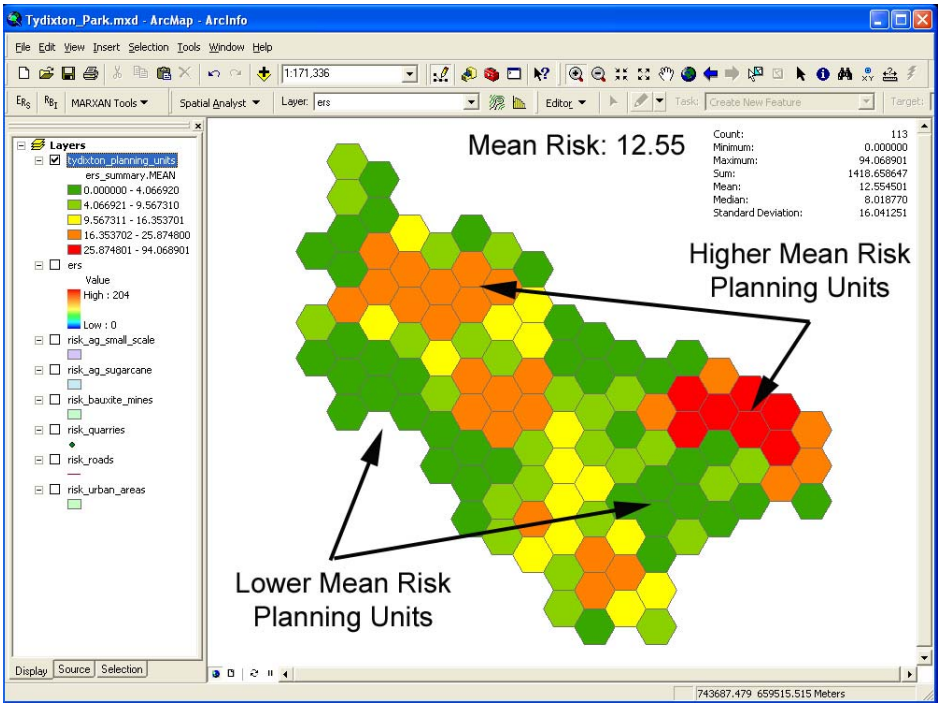
unit or feature (*i.e.* polygon) must have a unique ID field. If you do not have a unique ID field present, you must add a new integer field to the table and calculate the field to a unique value (*e.g.* FID field).

In this example, we will use the *tydixton_planning_units.shp* file to run zonal statistics of the ERS grid. This file was created using the HexGen tool listed under the Marxan Tools. Add this shapefile to the view and go to *Spatial Analyst* > *Zonal Statistics*. Make sure the following fields have been defined before clicking OK:

Zone dataset: *tydixton_planning_units.shp*
Zone field: *ID*
Value raster: *Your ERS ouput grid*
Ignore NoData in calculation: *checked*
Join output table to zone layer: *unchecked*
Chart statistic: *unchecked*
Output table: *specify the directory and name of the table. (e.g. zstat1.dbf)*

Once you click on OK, the surface univariate statistics of the ERS are computed within the boundaries of each unique planning unit hexagon. The *zstat* table will appear which can now be used to join back to the polygon shapefile by right-clicking on the planning unit shapefile and going to *Joins and Relates* > *Join.* When the Join Data dialog box appears, choose "*ID*" as the field that the join will be based on, make sure that the *zstat* output table is the table to join, and then choose the "*OID*" field in the table to base the join on. If you are using the habitat polygons and want to spatially equalize your ERS mean values, you may consider area weighting the ERS mean values by creating an area field (*e.g.* hectares or meters sq) in the habitat polygon attribute table and dividing the polygon mean value by the area of the polygon.

Once you have joined the table statistics to the planning unit shapefile, close the table and double click on the planning unit layer to bring up the *Layer Properties*. Click on the *Symbology* tab and specify *Quantities* > *Graduated color.* Choose the *Field Value* of "*zstat.MEAN*" (or the area-weighted



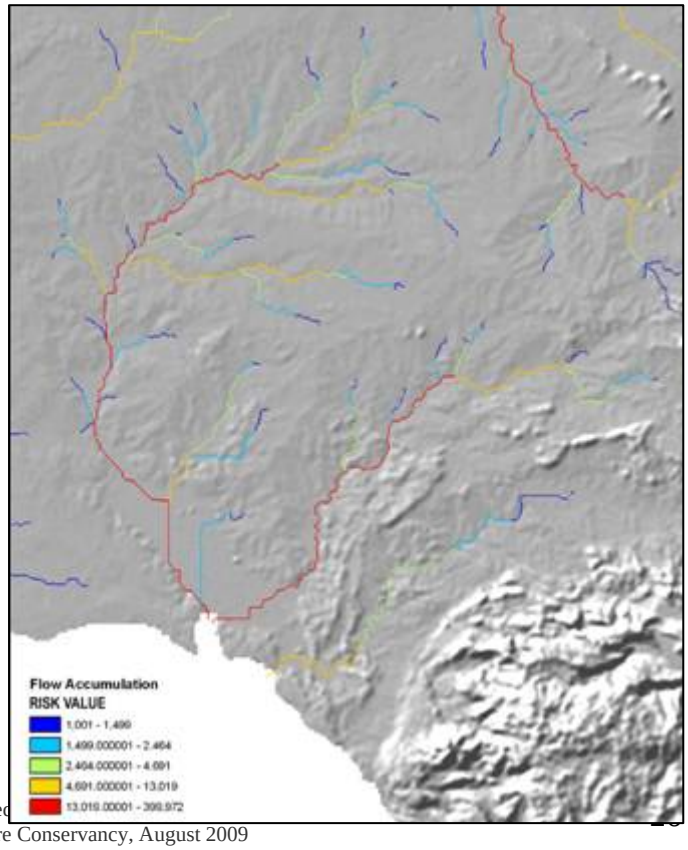
mean) and a color ramp. Before clicking OK, press the *Classify* button to see the MEAN statistic histogram summary. The mean value of the MEAN field will be displayed in the *Classification Statistics* box. This represents the mean risk value for all the habitat polygons. This number could be used as a cut off point in determining a threshold for habitats which may be more impacted because of the intensity and influence distances of the underlying risk element features. For example, if the mean value is 12.55, all planning units/polygons that exceed this may be high-risk candidate units, or potentially impacted at higher risk levels based on the ERS model and the spatial extents of the risk element features. One could exclude these features in the map view by clicking on *Data Exclusion* in the Classification dialog box, and specifying this threshold limit as a selection formula (*e.g.* "*zstat.MEAN*" > 12.55).

1.3.4 Creating Freshwater and Marine Environmental Risk Surfaces

As previously mentioned, ERS models can also be customized to represent risks to habitats in freshwater and marine realms. Experts in these realms must identify what risk elements are impacting the health of these systems and rank them accordingly in their intensity value and influence distance. For example, freshwater habitats may be impacted by features such as sewage outfall locations, dams, water abstraction, and invasive species. Marine habitats may be impacted by fishing pressures/practices, ports, marinas, and land- and marine-based pollution.

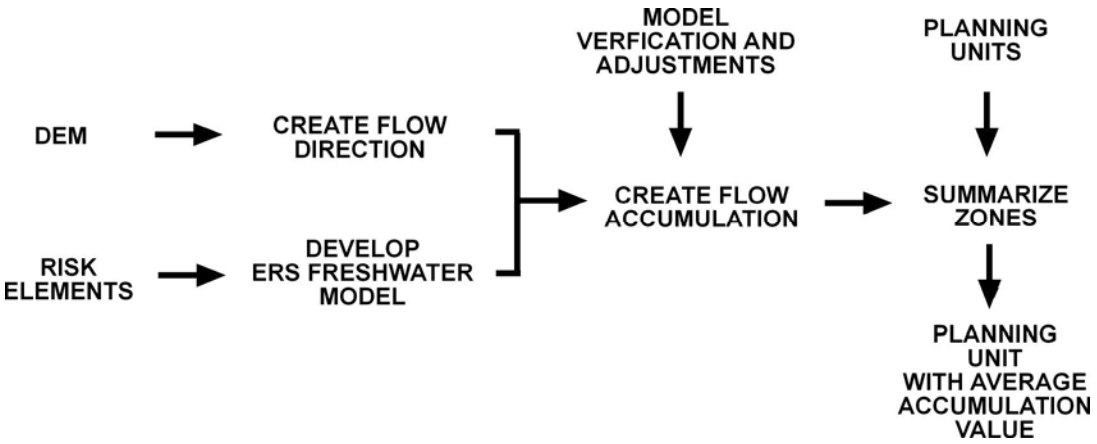
1.3.4.1 Freshwater ERS Models

A flow accumulation model can be used to measure the impact of defined risk elements on freshwater biodiversity by creating a grid of accumulated risk intensities that flow into each cell of a watershed. To do this, the user must first create an ERS model using expert-defined freshwater risk element features (as described above), and then specify the output grid model as the input source grid for the flow accumulation function (See FLOWACCUMULATION grid function in ArcGIS Help). This function requires the user to specify a flow direction grid which indicates the direction water flows on a cell-by-cell basis, starting at



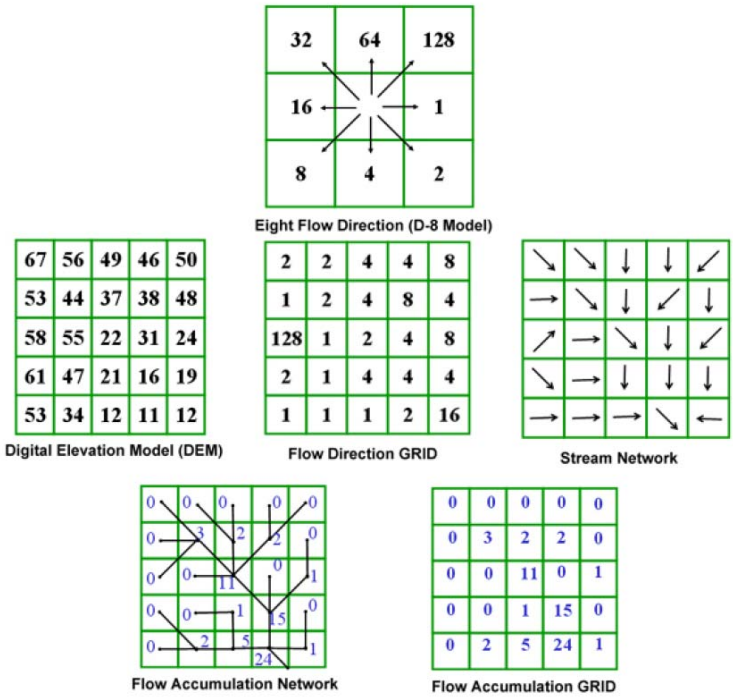
higher elevations and ridges moving downward, much like water moves across a sloped surface. In order to define a drainage network, the steepest down-slope flow path between each cell is identified between its eight neighbor cells.

Once the flow direction is established, the intensity values of the combined freshwater risk elements (*e.g.* agriculture, urban areas, and sewage outfalls) can be accumulated as the model runs from the top (ridges) to the bottom (outlet) of each watershed. The final accumulated grid surface calculates accumulated risk (in intensity values) upstream from any point (cell) in the watershed. Output cells with a high flow accumulation are areas of concentrated flow and may be used to identify stream channels and high risk areas. Output cells with a flow accumulation of zero are local topographic highs and may be used to identify ridges. This method also permits users to calculate the total risk intensity by watershed, thus quantifying the risk on a watershed level. If freshwater experts think that the high accumulated values are too high for large rivers, the effect can be lessened by subtracting a dilution factor to parts of the watershed. A zonal max function can be applied to locate the maximum flow accumulation value for each planning unit or unique watershed, thus quantifying the maximum flow accumulation value on a per unit basis.



Step flow process that uses a Digital Elevation Model (DEM) and Environmental Risk Surface (ERS) as a weight grid for accumulating intensity values when assessing risk to freshwater and marine habitats. In areas of low or zero slope, creating a suitable flow direction grid may be a challenge. Users may want to preprocess the DEM by "burning in" stream networks using a variety of tools available on the web (*i.e.* ArcHydro, AGREE). Rivertools is an exceptional hydrology modeling software with powerful tools for creating stream networks and ordered watersheds with high precision.

ArcHydro *http://www.crwr.utexas.edu/giswr/hydro/index.html*
AGREE *http://www.ce.utexas.edu/prof/maidment/gishyd97/terrain/agree/agree.htm*
Rivertools Software *http://www.rivix.com*


32 64 128
16 1
8 4 2
Eight Flow Direction (D-8 Model)
67 56 49 46 50
53 44 37 38 48
58 55 22 31 24
61 47 21 16 19
53 34 12 11 12
Digital Elevation Model (DEM)
2 2 4 4 8
1 2 4 8 4
128 1 2 4 8
2 1 4 4 4
1 1 1 2 16
Flow Direction GRID
Stream Network
Flow Accumulation Network
0 0 0 0 0
0 3 2 2 0
0 0 11 0 1
0 0 1 15 0
0 2 5 24 1
Flow Accumulation GRID

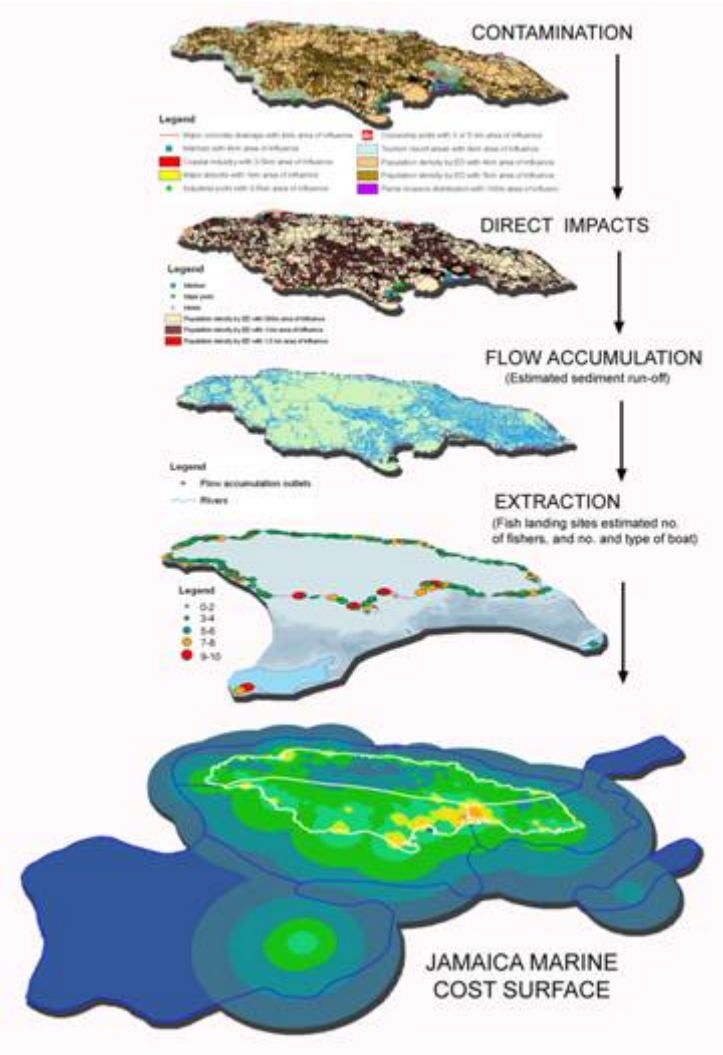
Process for creating flow direction and flow accumulation. Each cell can exit in one of eight directions. The flow direction grid is computed by the slope of exiting adjacent cells. Once the flow direction is established, the weight grid (*i.e.* intensity) can be accumulated throughout the network.

1.3.4.2 Marine ERS Models

Developing ERS models for the marine realm is often the most difficult of the three realms due to the lack of data and the dynamic nature of the ocean, an environment that is constantly moving. Assigning risk boundaries to the ocean is a complicated and perplexing task. Attempts to model coastal transport with ocean current data for assigning marine risk distributions has been attempted by Schill (2005) using the cell-assigned velocity from the Hybrid Coordinate Ocean Model (HYCOM) data model. The model operates using eastward and northward velocities on a cell-by-cell basis to compute a travel cost surface which represents the number of seconds it takes to cross a cell. Additional investigations are currently being conducting using the new network data model in ArcGIS Network Analyst.

As with the other, ERS models, marine experts must first identify the risk elements to the marine habitats. Human activities that are risk elements can be divided into four categories based on their marine impact: direct impacts (*e.g.* population density), contamination (*e.g.* coastal industry, ports/marinas), extraction (*e.g.* fishing and harvesting practices), and watershed-based sedimentation/run-off (land and marine-based pollution). This process was followed for the marine ERS models that were developed for the Jamaica Ecoregional Planning Marine Analysis (Zenny, 2006). Each risk element that was identified and mapped by experts was assigned an intensity value, an influence distance, then combined to create an overall risk surface.

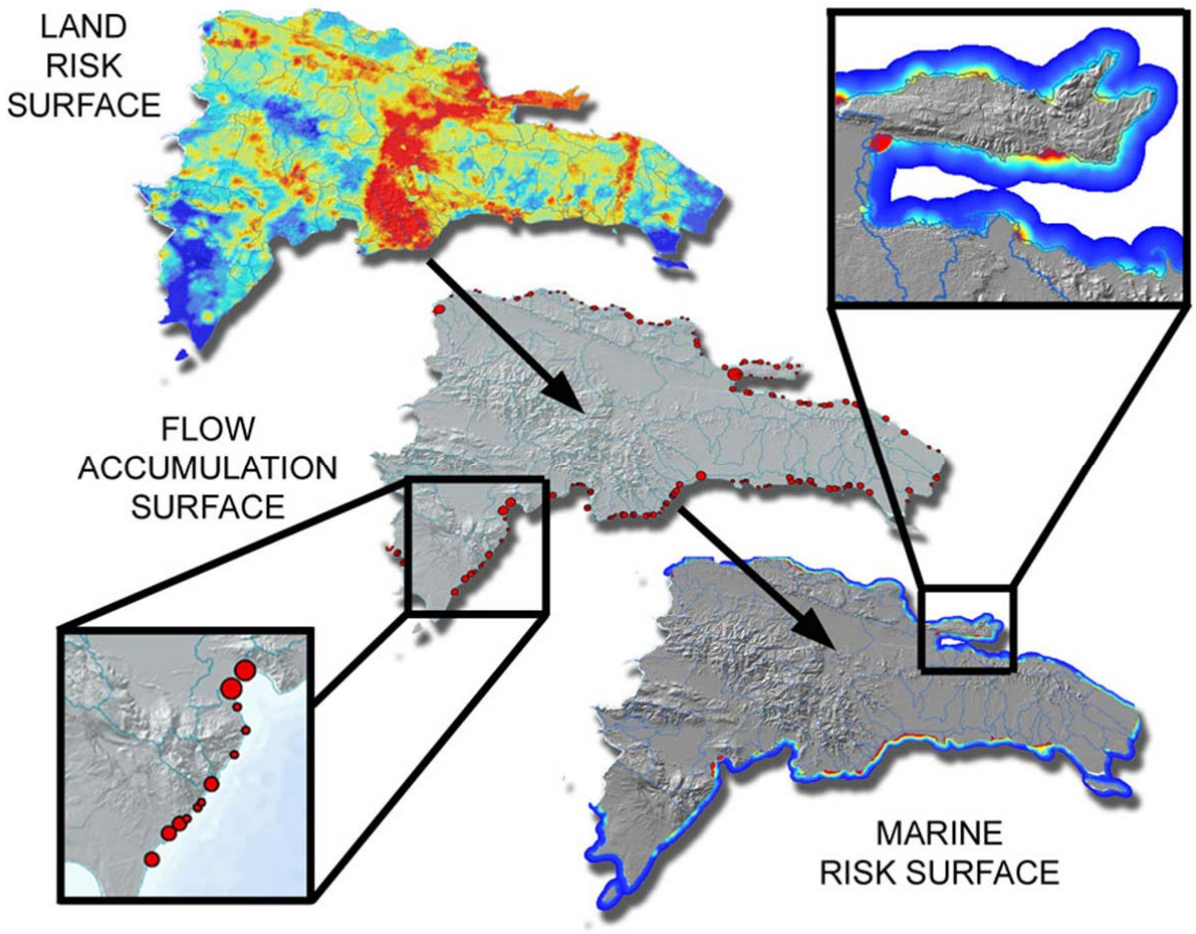
One idea for modeling potential upland sedimentation on coastal environments is to use the flow accumulation of the freshwater ERS model to gauge risk intensity at coastal outlets. As discussed in the previous section, the flow accumulation function aggregates and routes risk intensities to watershed outlets, which often empty into the



The creation of the Jamaican marine cost surface used in the MARXAN model included the combining of four risk categories: contamination, direct impacts, flow accumulation, and extraction.

ocean. GIS point features, representing coastal outlets, can be created at the end of the freshwater flow accumulated routes and the intensity values can be extracted and transferred to the point attribute table. The intensity values attached to these coastal points can then be assigned an influence distance and used as a part in the process of creating a comprehensive marine risk surfaces. The following ArcView 3.x tools were designed to extract a grid value to a point feature. A later version of the ERS module will include this functionality.

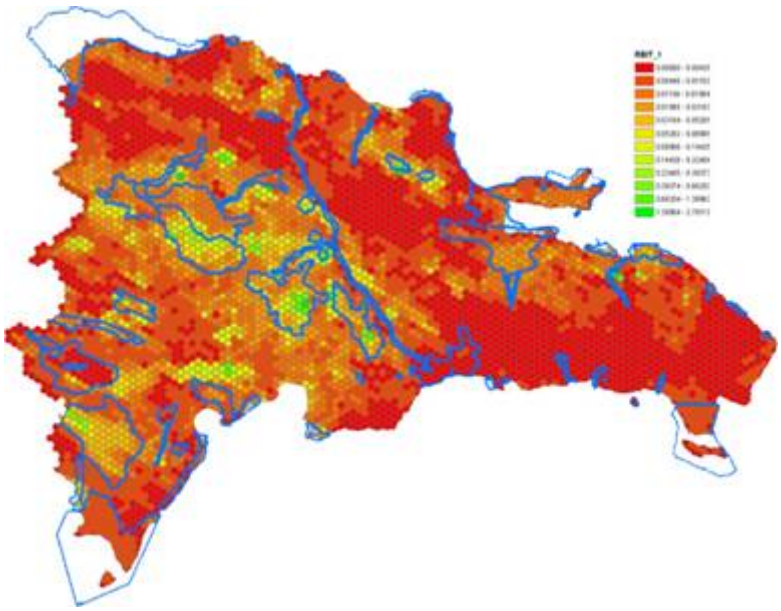
Get Grid Value Extension 2 *http://arcscripts.esri.com/details.asp?dbid=10200*
Grid Pig v2.6 *http://arcscripts.esri.com/details.asp?dbid=11872*



Using flow accumulation of land-based risk surfaces to estimate coastal outlet risk intensity on the marine environment in the Dominican Republic.

MODULE 2. Relative Biodiversity Index (RBI) Calculator

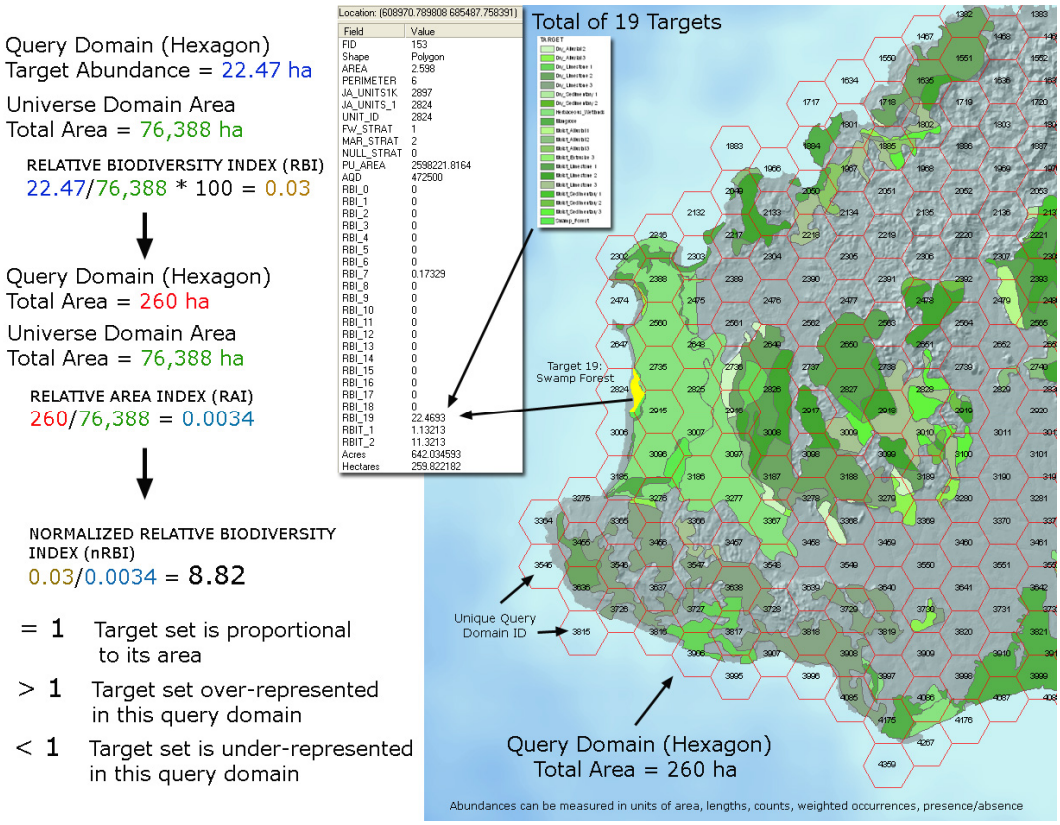
Although Marxan can identify an efficient portfolio that meets conservation goals for multiple conservation targets, in many cases, more target specific information is needed for management decisions. Marxan identifies the optimal solution, based on representation of multiple conservation targets. This optimal solution is often counter-intuitive and difficult to interpret and explain to managers and stakeholders. One of the reasons for this phenomenon is that Marxan is finding solutions for representation of *all elements* – and Marxan solutions represent a compromise between all the best areas for each target. Marxan solutions often exclude some of the best remaining areas for single targets and focus on areas that contain multiple targets. Many times, areas that are excluded from optimal solutions are intuitively important to protect.



As a complementary analysis to Marxan, a *normalized relative bio- diversity index* (nRBI) can be computed which quantifies the area-weighted relative contribution of each planning unit compared to the total distribution of each conservation target. In other words, the nRBI for each planning unit is directly proportional to the amount of conservation target present in the planning unit (*e.g.* hectares of habitat, length of stream or number of occurrences). The index can be summed for multiple targets, to create an aggregate nRBI. The advantage of this approach is that it can be used to identify the best remaining areas, in terms of target abundance, for each target or set of targets at the planning unit (*query domain*) or the landscape (*universe domain*) scale as shown in the figure below (TNC, 2005). In other words, you are calculating the relative uniqueness or rareness of a habitat or species across the landscape.

$$nRBI = \frac{RBI}{RAI}$$

where:
RBI = abundance (planning unit) / abundance (study area)
RAI = area (planning unit) / area (study area)


Query Domain (Hexagon)
Target Abundance = 22.47 ha
Universe Domain Area
Total Area = 76,388 ha
RELATIVE BIODIVERSITY INDEX (RBI)
22.47/76,388 * 100 = 0.03
Query Domain (Hexagon)
Total Area = 260 ha
Universe Domain Area
Total Area = 76,388 ha
RELATIVE AREA INDEX (RAI)
260/76,388 = 0.0034
NORMALIZED RELATIVE BIODIVERSITY INDEX (nRBI)
0.03/0.0034 = 8.82
= 1 Target set is proportional to its area
> 1 Target set over-represented in this query domain
< 1 Target set is under-represented in this query domain
Total of 19 Targets
Target 19: Swamp Forest
Unique Query Domain ID
Query Domain (Hexagon)
Total Area = 260 ha
Abundances can be measured in units of area, lengths, counts, weighted occurrences, presence/absence

Normalized Relative Biodiversity Index (nRBI) is calculated using an area-weighted function. When using *polygon* targets, values greater than 1 indicate a high level of habitat uniqueness when considering the overall landscape. The range and scale of RBI values will be different depending on what features are used as input (*e.g.* line, point).

This index computes *relative* abundance so the abundance can be any metric such as hectares of target, habitat, number of occurrences, length of stream, etc. It is important to note that the relative abundance calculations are different depending on the feature type of the target. For polygon targets which use area, values for nRBI greater than 1 indicate proportionately more target abundance in a planning unit than is expected for the planning unit size. Line targets use linear length and point targets have no area, so these features result in much lower values compared to polygon targets. Normalized relative biodiversity index values can be summed across multiple targets to calculate aggregate nRBI for terrestrial, freshwater and marine realms (see figure below). A higher RBI sum score (> 1 for polygon features) implies that there is a greater representation or extent of the targets than is expected for the planning unit size, which may or may not justify conservation action.

$$nRBI_T = \frac{\sum_{t=1}^{N} nRBI_t}{N}$$

where:
set of targets $T = [t_1, t_2, t_3,...,t_N]$
N = number of targets

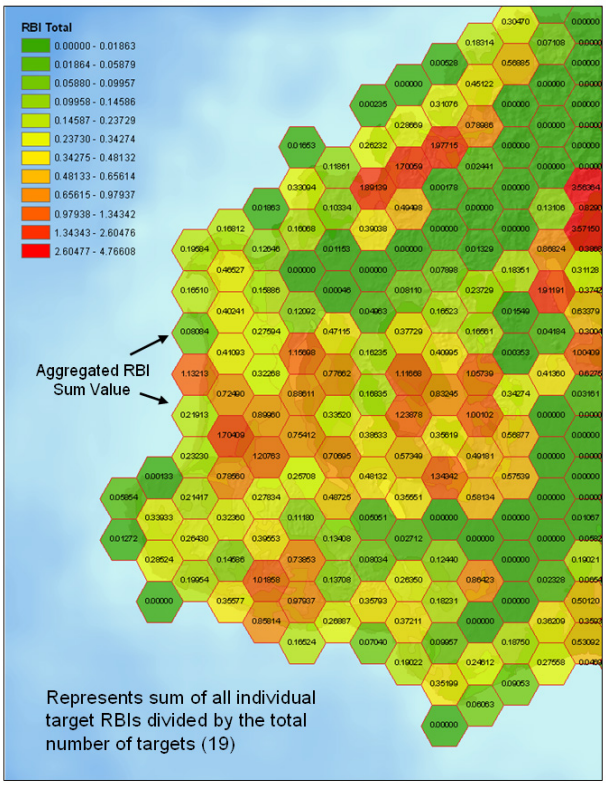
Aggregated Target Domains

Can sum up individual target RBIs across a set of targets T [$t_1 t_2 t_3 \ldots t_N$]

$$RBI_T = \left(\sum_{t=1}^{N} RBI_t \right) / N$$

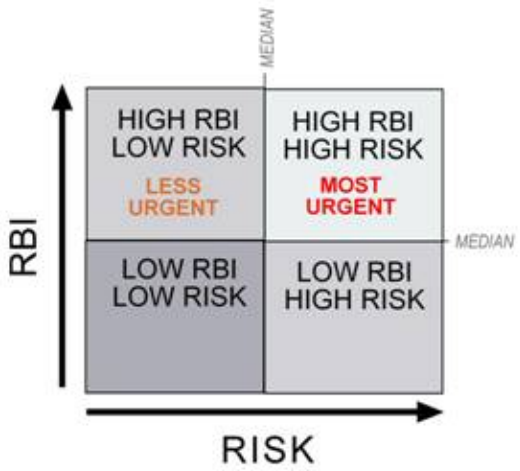
T can be defined as:
- All targets
- All forest matrix targets
- All riparian targets

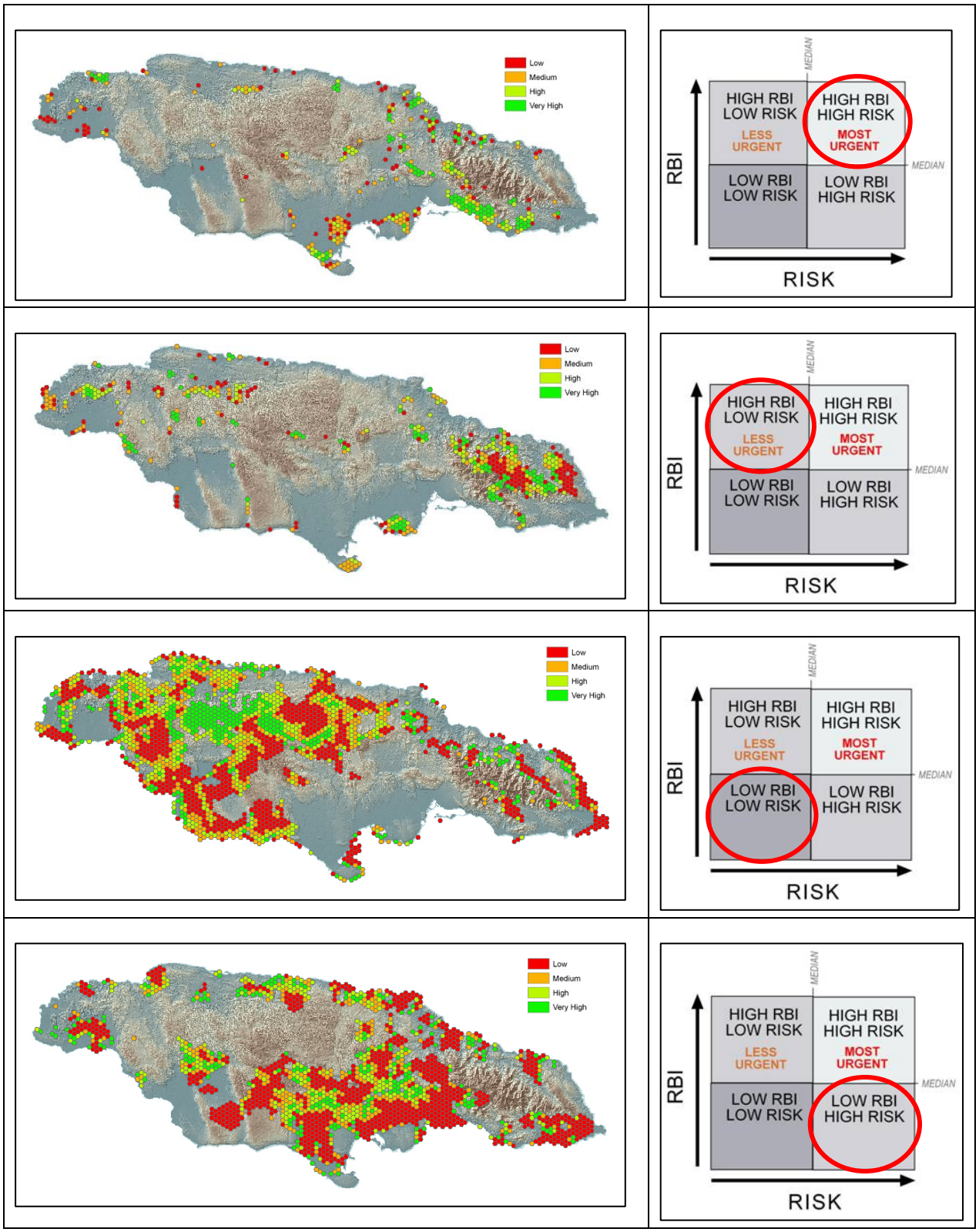
N can be defined as:
- The count of targets in the set T
- The count of targets in the set T that occur in that Query Domain



Normalized relative biodiversity index values can be summed across multiple targets (at the landscape and planning unit level) to calculate aggregate nRBI for terrestrial, freshwater and marine realms.

The Relative Biodiversity Index, calculated at the planning unit level, can be combined with the results of Marxan and environmental risk surface (ERS) modeling. This approach allows further, more specific, insight into potential conservation action such as setting priority sites that inform habitat-specific strategies. Moreover, this combination can be used to predict maximal return on conservation investment, towards long-term habitat goals and systematically provide sequence information for building a representative network of conservation areas. It is suggested that planning units containing both high nRBI (*i.e.* have relatively high target abundance) scores and high Marxan irreplaceability are areas that should receive first attention in sequencing conservation actions. These are areas that are potentially rich in rare, intact, or otherwise important ecosystem habitats. Additionally, rare habitat in high risk areas can be delineated by selecting areas above the median nRBI value and below the median risk (ERS) value. The figure on the next page shows an example RBI and risk analysis recently conducted for terrestrial habitats in Jamaica. Additional analysis can be done within strata to identify a more geographically distributed range of areas (TNC, 2005).





Results from an RBI and risk analysis for terrestrial habitats in Jamaica. This type of quartile analysis can be combined with Marxan output to guide prioritization of conservation areas. Areas are ranked from highest to lowest priority. High risk/high RBI areas may be more expensive to acquire but these habitats may be irreversibly lost if immediate action is not taken.

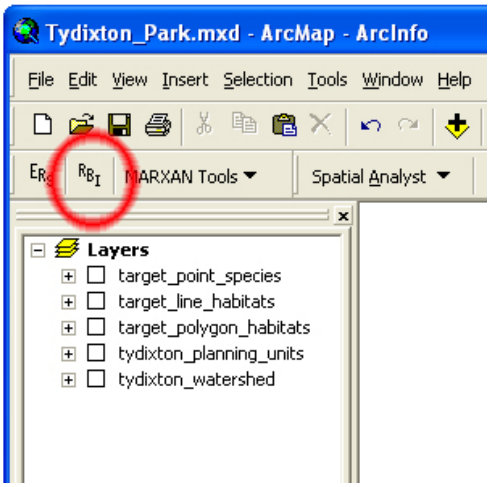
EXERCISE 2. Calculating the Relative Biodiversity Index (RBI)

REMEMBER

- All layers used in the analysis must be in the same projection and have the projection information defined. Do not use Geographic projection (e.g. decimal degrees) since unit areas or lengths will be calculated incorrectly.
- The RBI Module does NOT require an ArcGIS ArcInfo Level.
- The *Universe Domain* is the total study area or landscape (*i.e.* large watershed, political boundary)
- The *Query Domain* is the analysis unit or planning unit layer (*i.e.* hexagons or sub-catchment watersheds) and must have an ID field with a unique number assigned to each unit. A hexagon layer can be generated using "HexGen" in the Marxan Tools.
- The word "Target" refers to a conservation habitat or species and can be represented as a point, line, or polygon. When using your own data, make sure any entry in the target name field does not contain any special characters (*e.g.* ñ, ó, œ) or the program may fail.
- The target layer must have a unique name or ID field that represents each unique target class or type (*e.g.* forest type, stream type, species).

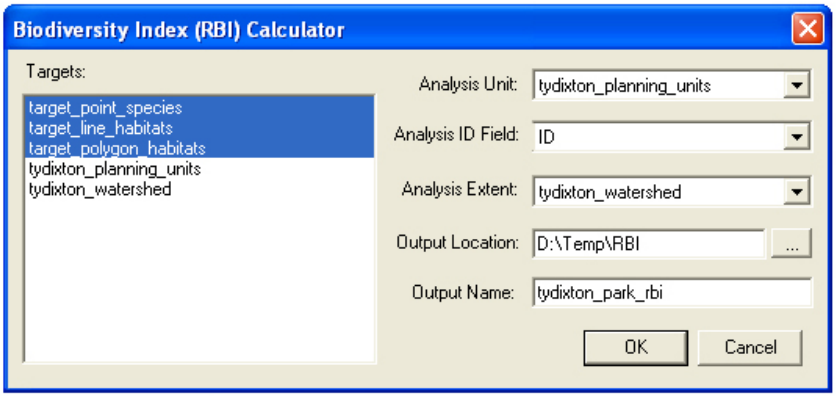
2.1 Running the RBI Module

The RBI tool requires three input data layers to operate on. The first layer is the *target layer*, or the layer that represents the conservation targets. These can be points, lines, or polygons. The second required layer is the *universe domain*, a polygon feature that represents the total analysis extent to be considered. This could be a country boundary or a watershed. The final layer needed is the *query domain*, also a polygon feature that represents the analysis or planning unit layer and the basic unit that receives the RBI calculation. This could be hexagons, or other polygon features such as watersheds.

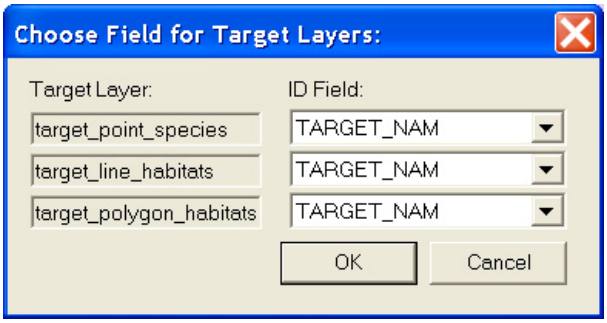


Prior to clicking on the RBI button, make sure you have the input files that will be used, loading into the view. For this exercise, this includes the target shapefiles (*target_point_species.shp*, *target_line_habitats.shp*, and *target_polygon_habitats.shp*); the Tydixton Park watershed boundary (*tydixton_watershed.shp*); and the analysis unit hexagons (*tydixton_planning_units.shp*). Remember that the analysis **extent** is also

referred to as the *universe domain* and the analysis unit is also called the *query domain*. The target layer must have a unique name or code field that represents each unique target class or type (*e.g.* forest type, stream type). The analysis unit layer must have an ID field with a unique code or ID assigned to each planning unit or watershed.



Once you have loaded the required layers in the map view, press the RBI button and the RBI Calculator dialog box will appear. In the left box, click on the targets files that you want to use in the RBI analysis. Remember this tool has been designed so you can run simultaneous calculations on point, line, and polygon features. For this exercise, choose *target_point_species*, *target_line_habitats*, and *target_polygon_habitats*. Next, choose the analysis unit or query domain layer from the drop down list (*tydixton_planning_units.shp)*. Next, choose the field from that layer that contains the unique ID for the analysis units (*ID*). In the Analysis Extent drop down list, choose the watershed boundary layer (*tydixton_watershed.shp*). The analysis extent is the universe domain (total landscape area) to be considered in the RBI calculations. Next, specify the Output Location as well as an Output Name. The output location is a folder, geodatabase or feature dataset. The Output Name you enter will serve as the root name of the two feature classes that will be created by the model. The two new feature classes that will be created in the Output Location, will be named "*your_output_name*_RBI_SUMMARY" and "*your_output_name*_RBI_ALL_TARGETS".



When you click OK, another box will appear that will ask you to specify the unique ID fields for each of the target input layers. For the three habitat shapefiles, you will choose the "*TARGET_NAM*" field. This field could also be a unique number code assigned to each target class or type, but it is easier to interpret the output tables if you use descriptive text for habitats and species such as "*Wet_Limestone_ Forest*," "*Karstic_Streams*," or "*Endemic_Fish*." When using your own data, make sure any entry in the target name field does not contain any special characters (*e.g.* ñ, ó, œ) or the program may fail.

When you have specified the target field for each of the input layers and press OK, the RBI calculations will begin processing. The process may run for quite some time, depending on number of analysis units, total number of unique targets, and complexity of

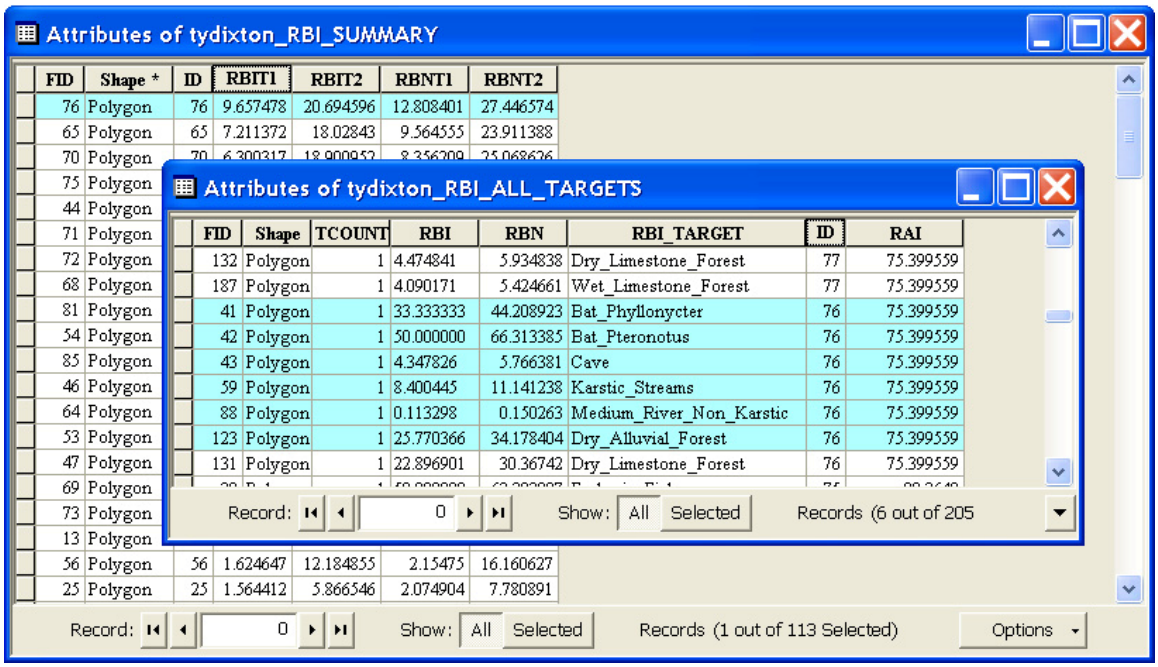
the targets (polygons with lots of vertices). It is best not to work on the computer or click on ArcMap while the model is executing. Once the model is completed, a text box stating "Processing Complete" will appear and the results will be loaded into the map view.

2.2 Interpreting RBI Results

When processing has completed, the *your_output_name*_RBI_SUMMARY feature class will appear in the map view's Table of Contents. The output is color ramped and displayed in an equal interval classification using the RBNT1 field, which is the aggregated Normalized RBI values summed across the total number of targets found within the *Universe Domain* (or total landscape extent). In this case, it is the Tydixton Park watershed boundary. The RBNT2 field is the aggregated Normalized RBI values summed across the total number of targets found within the *Query Domain* (or hexagon planning unit). As previously explained, when using polygon targets, higher RBI sum scores (> 1) implies there is a greater abundance or extent of the targets than is expected for the analysis unit size, in comparison to the rest of the landscape. The range and values of the RBI scores will be different depending on the combination and number of feature types you use. Open the Attribute Table by right-clicking on the *your_output_name*_RBI_SUMMARY layer and you will see a series of RBI columns:

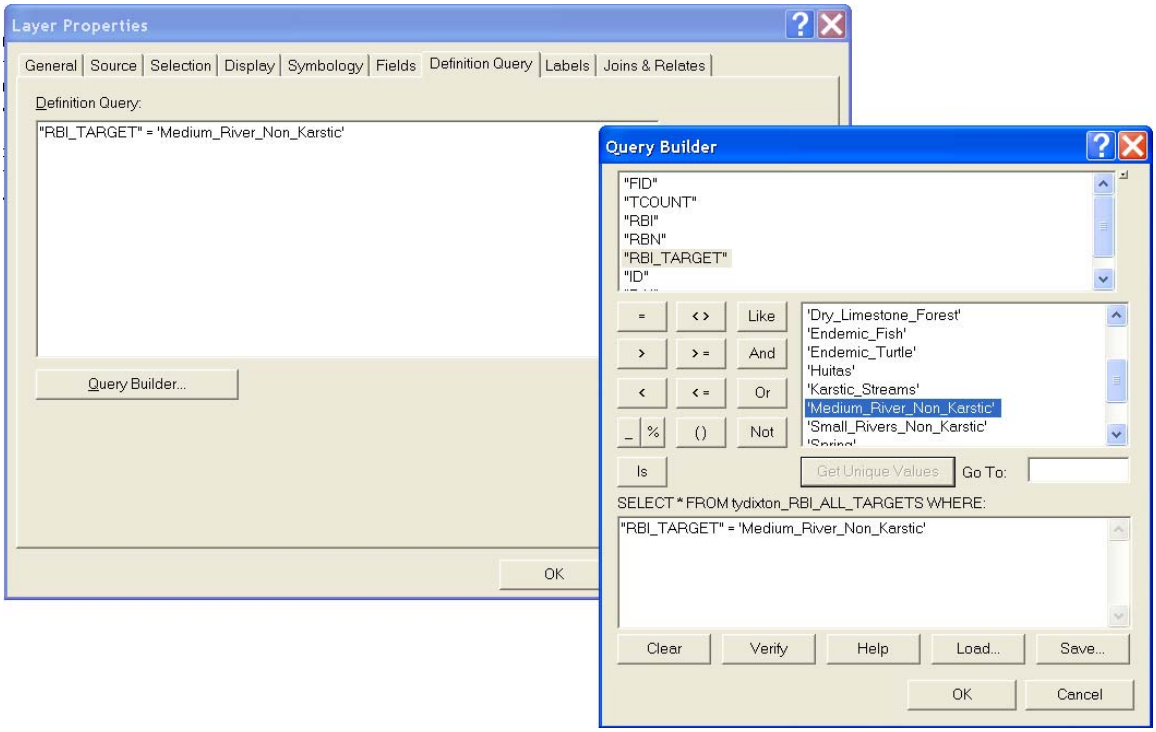
FIELD	*DESCRIPTION*
RBIT1	**Non-normalized** RBI value average based on the sum of total number of targets found in the *universe domain* (*i.e.* total landscape).
RBIT2	**Non-normalized** RBI value average based on the sum of total number of targets found in the *query domain* (*i.e.* hexagon planning unit)
RBNT1	**Normalized** RBI value average based on the sum of total number of targets found in the *universe domain* (*i.e.* total landscape).
RBNT2	**Normalized** RBI value average based on the sum of total number of targets found in the *query domain* (*i.e.* hexagon planning unit)

As part of the RBI output, another feature class is created called "*your_output_name*_ RBI_ALL_TARGETS." If we look at the RBI model output, planning unit number 76 came out the highest in the RBNT1 score. If we open the "*your_output_name*_RBI_All_ TARGETS" table we can see the list of targets and their associated RBI scores within planning unit 76:

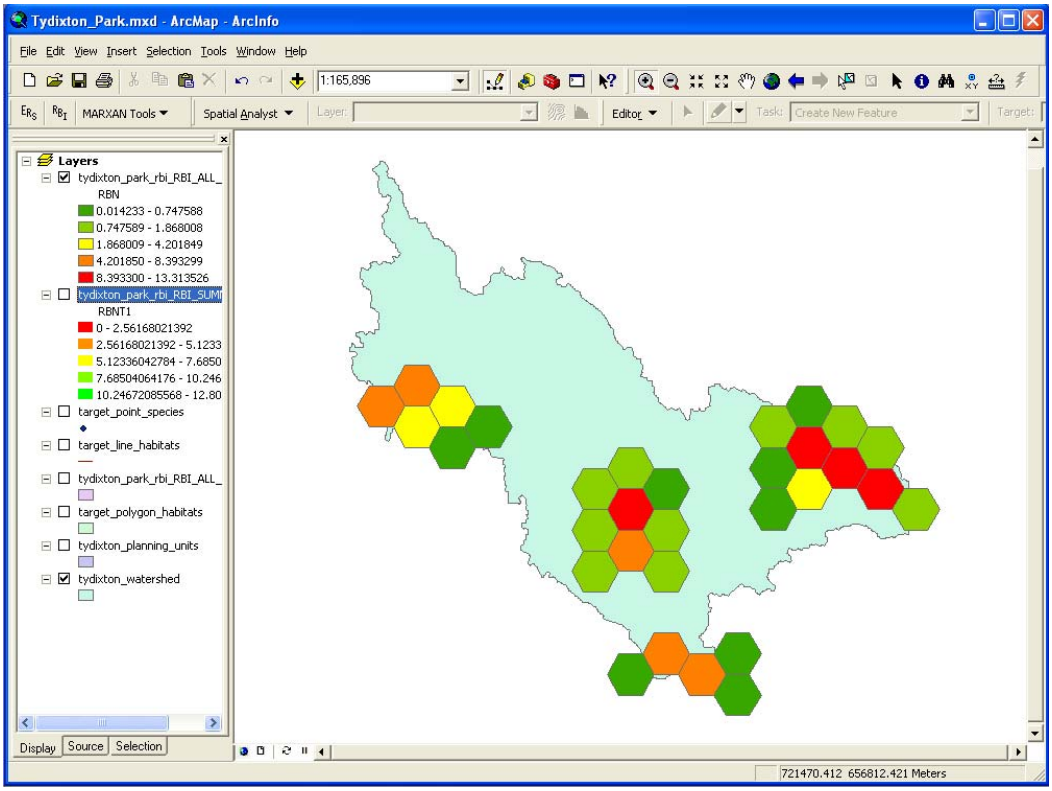

Attributes of tydixton_RBI_SUMMARY
FID Shape * ID RBIT1 RBIT2 RBNT1 RBNT2
76 Polygon 76 9.657478 20.694596 12.808401 27.446574
65 Polygon 65 7.211372 18.02843 9.564555 23.911388
70 Polygon
75 Polygon
44 Polygon
71 Polygon
72 Polygon
68 Polygon
81 Polygon
54 Polygon
85 Polygon
46 Polygon
64 Polygon
53 Polygon
47 Polygon
69 Polygon
73 Polygon
13 Polygon
56 Polygon 56 1.624647 12.184855 2.15475 16.160627
25 Polygon 25 1.564412 5.866546 2.074904 7.780891
Record: 0 Show: All Selected Records (1 out of 113 Selected) Options
Attributes of tydixton_RBI_ALL_TARGETS
FID Shape TCOUNT RBI RBN RBI_TARGET ID RAI
132 Polygon 1 4.474841 5.934838 Dry_Limestone_Forest 77 75.399559
187 Polygon 1 4.090171 5.424661 Wet_Limestone_Forest 77 75.399559
41 Polygon 1 33.333333 44.208923 Bat_Phyllonycter 76 75.399559
42 Polygon 1 50.000000 66.313385 Bat_Pteronotus 76 75.399559
43 Polygon 1 4.347826 5.766381 Cave 76 75.399559
59 Polygon 1 8.400445 11.141238 Karstic_Streams 76 75.399559
88 Polygon 1 0.113298 0.150263 Medium_River_Non_Karstic 76 75.399559
123 Polygon 1 25.770366 34.178404 Dry_Alluvial_Forest 76 75.399559
131 Polygon 1 22.896901 30.36742 Dry_Limestone_Forest 76 75.399559
Record: 0 Show: All Selected Records (6 out of 205

Based on the model results, we see that almost all the RBN scores for this planning unit were calculated above 1. This indicates there is a high abundance of unique target occurrences/areas contained within this planning unit compared to the same target(s) occurring in the rest of the Tydixton Park watershed. Remember that these scores are relative and their range of values can vary widely depending on the input data that is used. The highest two scores were for two rare bat species where there are very few occurrences throughout the watershed. Similarly, you will notice that Dry Alluvial Forest is a very rare habitat type and it occurs only in small amounts within the watershed (total of 38.8 hectares). If you look at planning unit 65, you will notice that a large majority of what does exist for this habitat is located in that unit (approximately 28.79 hectares). This was the highest individual RBN score in the entire watershed (73.87). Similar conclusions can be made for planning unit 65 where *Wet_Alluvial_Forest* and *Small_Rivers_Non_Karstic* also scored relatively high RBN values (34.06 and 21.64, respectively). Please note that if you are only using point feature class targets, your RBI scores will be significantly lower, given the fact that there is no area to calculate for points.

You can choose to symbolize the RBI values for a single Target using the "*your_output_name*_RBI_ALL_TARGETS" feature class. To do this, go to the layer's properties dialog and select the "Definition Query" tab. Then use the query builder to create a query that will only display the information for the target you are interested in. See the example below.

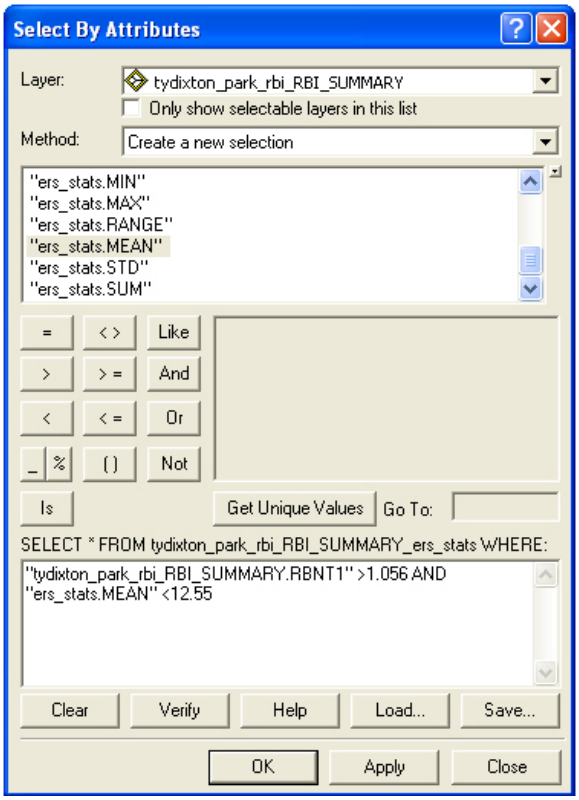


Once this step is performed, you can symbolize this layer using the "symbology" tab in the layer properties.



Classified RBI values based solely on the "Medium_River_Non_Karstic" streams target.

We are now ready to add the statistical surface summaries of the Environmental Risk Model (ERS) and begin to identify planning units that have high RBN values (above the mean) and low risk values (below the mean). These planning units could be candidate sites for high conservation value that are in relatively low-risk environments, according to our modeled ERS surface. If you have not already run *Zonal Statistics* using a modeled ERS surface, follow the instructions in the first module, using the *your_output_name*_RBI_SUMMARY output shapefile as the Zone dataset and join the output statistics table back to the *your_output_name*_RBI_SUMMARY output shapefile.

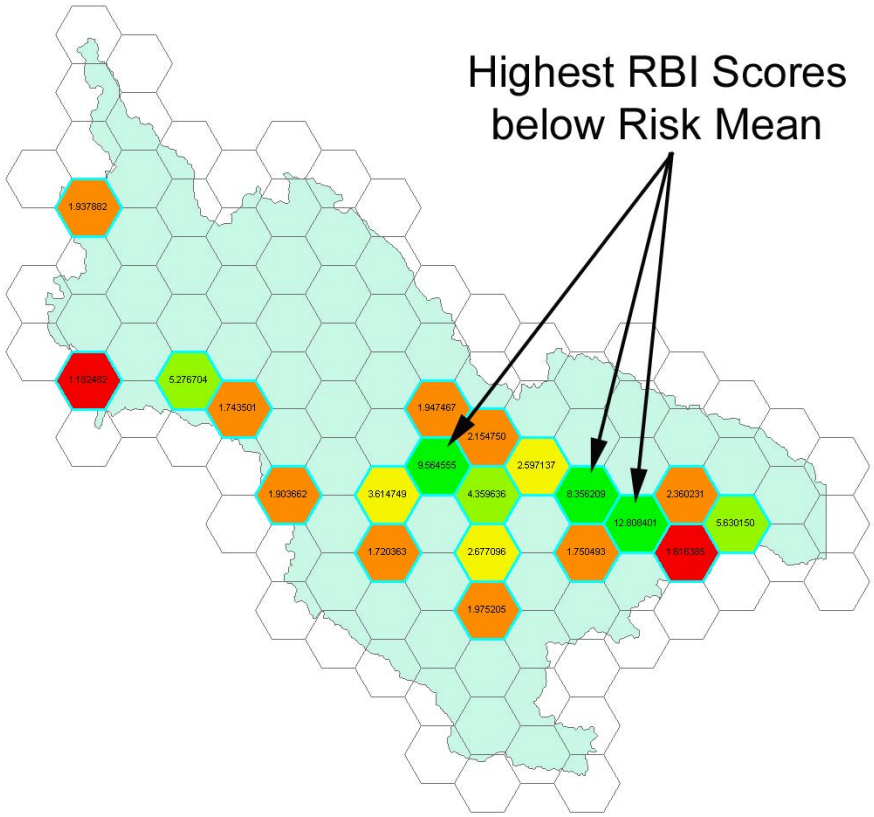


Now determine what the mean values are for both the ERS surface and the RBI output shapefile. This can be done by double clicking on each layer and going to *Symbology* > "*Classify*" or "*Histogram*" option. In either option the *mean* for the dataset will be reported. For the ERS surface, the mean is *12.55* and for the *your_output_name*_RBI_SUMMARY output, the mean is *1.056*. Now we will use these numbers to threshold out high RBI pockets that have low risk value.

Go to *Selection* > *Select by Attributes* in the main ArcGIS toolbar. When the dialog box appears, enter the following formula:

" *your_output_name*_RBI_SUMMARY.RBNT1" > 1.056 AND "ers_stats.MEAN" < 12.55

This selection routine will choose the planning units (highlighted in light blue) that are above the mean RBI value and below the mean risk value, indicating high conservation value with low risk based on model inputs as shown in the figure below. This approach is an example of providing insight into potential conservation areas and setting priority sites based on target-specific strategies for maximum return on conservation investment. Similar analysis can be conducted using Marxan irreplaceability (number) values.

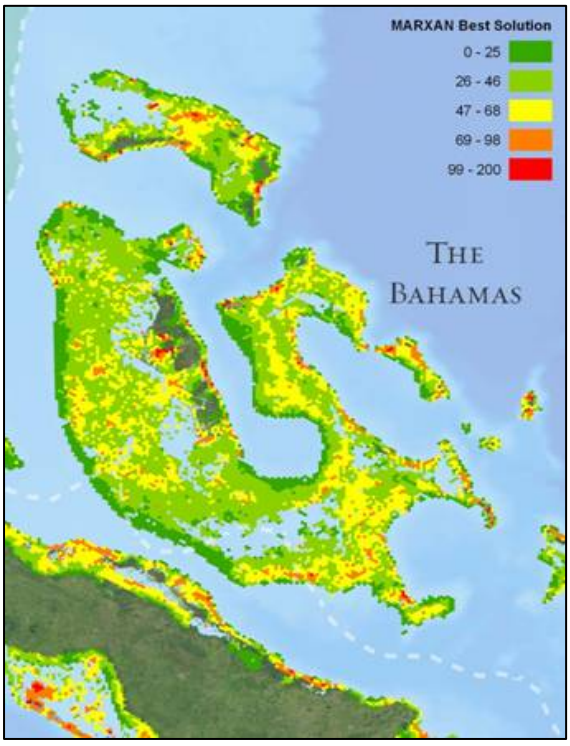


An example of selecting planning units (highlighted in light blue) that are above the mean RBI value (> 1.056) and below the mean risk value (< 12.55). These planning units indicate areas of high conservation value within low risk areas, based on model inputs.

MODULE 3. Marxan Tools

Marxan is a free software program that provides decision support to teams of conservation planners and local experts identifying efficient portfolios of planning areas that combine to satisfy a number of ecological, social, and economic goals (Ball and Possingham, 2000). Marxan is used by over 1100 registered users from 600 organizations (government, academia and NGOs) in 95 countries as a decision support tool to consider options in terrestrial and marine reserve design. It is perhaps the most popular "site selection optimization" software available and is widely published. It is a stand-alone program that requires no other software to run, although GIS makes it simpler to prepare the data, generate the required input files, and view the results. For the latest information about Marxan, users should register and download the software at http://www.uq.edu.au/marxan

Early conservation assessments depended on manual mapping to delineate sites and were often reliant on expert opinion to prioritize conservation areas. The large number, size, and diverse types of datasets describing the targets eventually required the use of a more systematic and efficient site selection procedure. Marxan software is an optimization program and provides decision support for teams of experts choosing between hundreds of biodiversity targets and thousands of candidate areas (planning units). It identifies efficient portfolios of planning units and has a measure of flexibility that allows the teams to adapt efficient solutions to real world situations. Using a transparent process that is driven by quantitative goals, the analysis is repeatable and objective. Marxan results can illustrate a pattern of priority sites of low political or social pressure that can still satisfy the explicit biodiversity goals. It can also identify a network of sites where resources necessary to implement conservation strategies or threat abatement are forecast to be lower (TNC, 2005).



Planning units are parts of the land and seascape that are analyzed as the potential building blocks of an expanded system of reserves or areas of conservation priority. They allow a comparison between candidate areas. Planning units can be systematic units such as hexagons; natural areas like watersheds, administrative boundaries; or arbitrary sub divisions of the landscape. They differ widely in size between studies and within regions, dependant mostly on scale of analysis and data resolution (TNC, 2005).

3.0 The MARXAN Algorithm

One of the more popular algorithms implemented in Marxan is the '*simulated annealing*' site optimization algorithm. In order to design an optimal reserve network, each planning unit is examined for the values it contains. The features within one planning unit may be valuable alone but may not be the best choice overall—depending on the distribution and replication of the other features in the wider planning area.

During the simulated annealing procedure, an initial portfolio of planning units is selected. Planning units are then added and removed in an attempt to improve the efficiency of the portfolio. Early in the procedure, changes in the portfolio that do not improve efficiency can be made in order to allow the possibility of finding a more efficient overall portfolio. The requirement to accept only those changes that improve efficiency becomes stricter as the algorithm progresses through a set of iterations. Note that for any set of conservation targets and goals, there may be many efficient and representative portfolios that meet all conservation goals, but most of these networks would have a number of planning units in common. Many runs of the algorithm are used to find the most efficient portfolio and to calculate a measure of *irreplaceability* (used here to indicate the number of times a particular unit is chosen). In some cases, conservation targets are only found in limited sites – areas of high irreplaceability that are always chosen in any representative portfolio. Additionally, areas of high irreplaceability also include planning units, whose exclusion would require a proportionally larger conservation area network to achieve the same level of representation, resulting in a loss of portfolio efficiency.

The algorithm attempts to minimize portfolio total '*cost*' whilst meeting conservation goals in a spatially compact network of sites. This set of objectives constitutes the '*objective cost function*' and is made up of three user-defined 'costs':

$$\textbf{\textit{Total Cost} = } \sum \textbf{\textit{Unit Cost} + } \sum \textbf{\textit{Species Penalties} + } \sum \textbf{\textit{Boundary Length}}$$

where '*Total Cost*' is the objective (to be minimized), '*Unit Cost*' is a cost assigned to each planning unit, '*Species Penalties*' are costs imposed for failing to meet biodiversity target goals, and '*Boundary Length*' is a cost determined by the total outer boundary length of the portfolio.

Attempts are made to minimize the total portfolio cost by selecting the fewest planning units with the lowest total unit cost needed to

meet all biodiversity goals, and by selecting planning units that are clustered together rather than dispersed (thus reducing outer boundary length). This task is accomplished by changing the planning units selected and re-evaluating the cost function, through multiple iterations. Alternative scenarios can be evaluated by varying the inputs to the total cost function. The boundary length cost factor, for example, can be increased or decreased depending on the assumed importance of a spatially cohesive portfolio of sites (TNC, 2005). Conservation portfolios can be identified that met stated goals for representation of the biodiversity targets. The ultimate objective is to find a portfolio that meets stated conservation goals for all target groups in an efficient manner, while also meeting the general criteria of reserve design (*e.g.*, connectivity, minimal fragmentation). While this tutorial does not cover all the aspects of Marxan, additional information can be found in Ball and Possingham, (2000).

New Marxan User Manual (For Marxan version 1.8.10)
The Pacific Marine Analysis and Research Association (PacMARA) and the University of Queensland have written an enhanced user manual for Marxan that was released in February 2008 (Game and Grantham, 2008). Previous documentation was a mixture of mathematical theory and technical input data requirements and did not fully meet the needs of all tool users. It is highly recommended that all Marxan users download this manual from http://www.uq.edu.au/marxan/index.html?page=77823&p=1.1.4.2 and review with detail since it greatly facilitates understanding of the core software well beyond what this tutorial is designed for.

EXERCISE 3: Running a Sample Marxan Analysis

REMEMBER

1. The user needs to have installed **ArcGIS 9.3** ArcMap (9.0, 9.1, or 9.2 may not work. Also please make sure you have installed the latest service pack**)**
2. An **ArcInfo Level** is required to run the Marxan Tools. In addition, you also need to have **ArcInfo Workstation** installed and make sure the **Coverage Tools Toolbox** is loaded (this is the default). This toolbox is usually found in C:\arcgis\arcexe9x\Toolboxes (If the user has installed it to the default location)
3. The user must have the **Spatial Analyst** Extension and it should be **turned on** (Tools > Extension)
4. The user should have all **geometry checked/repaired** for each input feature file. If problems persist after repairing geometry, converting shapefiles into coverage format, then back to shapefile format will often solve the problem. The Marxan Tools will process geodatabases and shapefiles but NOT grids.
5. It is highly recommended to use the Target Prep tool, since it creates all the necessary fields used in the Marxan Input Generator and dissolves each target (*i.e.* there is one table listing for each unique target type).

6. All input features must have the **projection defined** and all input features must be in the **same projection.** Do not use Geographic projection (e.g. decimal degrees) since unit areas or lengths will be calculated incorrectly.
7. Remember that depending on the size and level of detail requested by the user, the Marxan Tools **may require large amounts of disk space** to operate and may take a **considerable amount of time to run**.
8. You may want to temporarily **disable any Virus Checkers** since the program will be accessing and deploying executable files.
9. When using your own data, make sure any entry in the target name field does not contain any special characters (*e.g.* ñ, ó, œ) or the program may fail.
10. If you have not already, users should register and download Marxan from http://www.uq.edu.au/marxan . You need around 2 MB of free disk space to install Marxan and the associated files. When you download Marxan you will receive the following files:
    a. Marxan.exe (the Marxan program executable)
    b. Inedit.exe (a program that allows you to easily generate the Input Parameter File – the file that controls how Marxan works)
    c. input.dat (an example Input Parameter File)
    d. A folder labeled 'Sample', containing examples of the other input files used to run Marxan
    e. The Marxan User Manual

    These files can be saved anywhere on the computer. For simplicity when running Marxan, the executable, 'Marxan.exe', should be located in the same folder as the input files for that project. Rather than continually moving files around, Game and Grantham (2008) recommend simply copying the Marxan executable to each folder containing a Marxan project.
11. It is important to note that there are a number of Marxan variations with modified functionalities that have been developed over the years (Game and Grantham, 2008). There is a version that allows probabilistic information on threats or the presence of conservation features at sites to be included in the reserve design problem; and Marxan with Zones, which is being developed to handle multiple objective zoning The most widely used version is Marxan 1.8.10 which uses a traditional tabular matrix format. Users should also be aware of an optimised version 2.0.2 which uses only uses a sparse matrix format and was redesigned to handle very large and complex problems involving greater than 20,000 planning units. This version is sensitive to the order of planning unit identifiers in the planning unit by species sparse (or relational) matrix and is not compatible with the traditional tabular matrix format. Please consult the Marxan website (http://www.uq.edu.au/ marxan) to obtain a copy of the command line program "*convert_mtx.exe*" which converts existing Marxan tabular matrix files into a format compatible with Marxan Optimised.

3.1 Setting the Scenario and Marxan Project Considerations

Before executing a Marxan model, conservation experts should decide on scenario settings which may include a) *the set of data to be used*; and b) *the conditions applied to the objective function for Marxan to use* (Ball and Possingham, 2000). Users must feel comfortable about the data that is going into the model and the conservation goals that have been assigned. It is a difficult and often abstract task to set goals, decide on cost measures, planning units, etc., but these are essential decisions that must be made. Remember that Marxan produces flexible solutions that meet specific quantitative representation goals in addition to minimizing threats constraints. As this is an iterative process, a number of conservation scenarios can be formulated and explored using Marxan. There are also a number of ways solutions can be tested and found using the heuristics or the annealing algorithm.

In addition to setting the conservation scenarios, there are several project considerations to be made before finalizing the conservation targets, planning units, protected areas, and cost surface. Ideally, these considerations should be made through consultation with conservation experts and some can be determined through experimental model runs. The following are suggestions from Huggins (2005):

a. ***Analysis Extent***: The areas of interest should be defined based on the extent of all target and socio-economic data that have been gathered and assessed. The extent should be defined by a polygon so that all features to be used in the model are clipped to the extent of the analysis area.

b. ***Target Screening***: All target distributions to be entered into the analysis should be considered viable occurrences that are robust enough to influence the portfolio selection. Screening biodiversity distribution maps should be considered to improve the likelihood of only including viable occurrences. Methods previously used include screening vegetation by expert opinion, removing all patches below a threshold size, and screening using threat factors such as an Environmental Risk Surface (ERS) or freshwater flow accumulation models.

c. ***Target Stratification***: Targets can be stratified to allow a geographic spread of representation or to represent biologically distinct target sub-groups. The portfolio will be greatly influenced by the way the targets are defined. For example, if the portfolio is required to hold both upland and lowland portions of a specific target, it must be defined in a way that includes this information. Examples include stratification by elevation or bathymetry, surface geology, or by other geographic units that define biologically meaningful differences.

d. ***Designing Planning Units (Adapted from Game and Grantham, 2008 and Pressey and Logan, 1998):*** An essential pre-processing step is to divide your planning region into a set of planning units. Planning units are areas for which data on occurrence, frequency and extent of the targets exists. In their simplest form, planning units may be defined by overlaying your planning region with a

grid of squares or lattice of hexagons. They must capture all the areas that can possibly be selected as part of the reserve system and their size should be at a scale appropriate for both the ecological features you wish to capture and the size of the protected areas likely to be implemented. In general, they should be no finer in resolution than the data on conservation features and no coarser than is realistic for management decisions. There is, however, no necessity to have uniformly shaped planning units. Nor is it always true that smaller planning units are better. In some cases it will make more sense to have planning units that are informed by natural ecological divisions such as hydrological units, or even by political/governmental divisions such as cadastral parcels. For other uses, a uniform planning unit will provide more useful results.

The choice of planning units has important implications for the process of portfolio selection as well as implementation of its results. The choice of planning unit size and configuration for both wide and local scale analysis must be made with many factors in mind. These include:

- The *size of the planning unit relative to the scale of the underlying features* (*e.g.* planning units that are much larger than underlying fragments of vegetation can mask the size, shape and extent of fragmentation; planning units that are very small relative to the vegetation types will mostly be homogeneous, *i.e.* small and large patches of habitat will be indistinguishable).
- The *number* of planning units that can be handled by the analysis computer in a time that is reasonable for the intended process (*e.g.* calculation of target area within planning units, clustering test). There is a limit on the number of planning units that Marxan 1.8.1 can handle. This is not, however, a fixed number as it depends also on the number of conservation features you wish to plan for and even to some extent on the power of your computer. Marxan 2.0.2 can handle very large marxan projects (>20,000 planning units) and is really only limited by available memory.
- The *size of planning units in relation to the reliability of mapping* (*e.g.* larger planning units could be needed where the locations of the targets to be represented are imprecise or where the boundaries of planning units are known to be inaccurate).
- The ability of regular grids or hexagons to *show per unit area values for criteria* such as richness of unprotected targets.
- *Equality of the sizes of planning units over large geographic areas* when factors such as map projections are an issue.
- *Convenience of conversion of planning units to management units* on the ground when analyzing at the fine scale.
- *Appropriateness of boundaries for conservation management* when analyzing at the fine scale.

e. ***Planning Unit Cost Range***: The range of cost values must allow the desired influence on the selection of planning units. This must be considered with relation to boundary length, as the objective function is a combination of boundary length, boundary length modifier (BLM) and cost. The effect of the BLM will also be affected by changes in cost. If a range of costs is to be used, the BLM value should be tested with the costs in place. The Species Penalty Factor (SPF) is the importance a user puts on each target meeting its goal and also influences the selection of planning units. All of these factors have to be in correct balance to arrive at an optimal portfolio, so the user has to decide how much influence they would like each factor to have. Some planning situations require the cost surface to have a larger or smaller effect, and the same can be said about the amount of clustering needed. The amount of experimentation on the cost range with the BLM depends on the magnitude of the boundary in comparison with the importance of each target meeting its goal.

There are three suggested scenarios users should try pertaining to cost range (Huggins, 2006):

1. Set a "flat cost" of all units set to the area of the planning unit (*e.g.* 260)
2. Set a range of cost values with 260 as base, then going higher with the cost factors (*i.e.* 5* base cost) so you end up with a fairly tight range (*e.g.*, 260 - 1300).
3. Set cost where managed protected areas (*e.g.*, MPAs) offset the cost, so that the cost in these areas is reduced by a factor (*e.g.* 0.33). In this case, if you have a planning unit where trawling activities drive the cost up, the managed protected areas will have a lower unit cost. That way there is a quantitative difference between trawling in and out of a managed area. This should produce better results than locking in managed protected areas as a scenario.

f. ***Boundary Length Modifier (BLM) Experimentation***: The BLM is specified in the input parameter file using Inedit.exe. Before the boundary length of a portfolio is added to the Marxan cost function (as the boundary cost), it is scaled by the BLM. There is no theoretically good value to give it because the cost measure and the length measure are both arbitrary (Ball & Possingham, 2000). The BLM can be confusing to understand, but serves several important purposes:

- To specify the relative importance of fragmentation in the cost function. Smaller values will make fragmentation less important than meeting goals and minimizing area. If the BLM is set to zero, then the boundary length will have no impact on the selection of the portfolio, and the output solution will be highly fragmented. The more you increase the BLM, the more you increase the importance of compactness for the modeled portfolio solution.
- To convert units. If the base analysis unit cost is specified as hectares and boundary length as kilometers, the BLM must serve the purpose of converting the boundary into comparable units.

- To make "area" and "length" comparable. The least fragmented shape possible is a circle, and the area to circumference ratio can serve as a guide for this.

Because of the many conflicting factors inherent in the BLM, the best way to arrive at a good number is via experimentation. There are some key things to look at to test the BLM's "accuracy" at meeting goals, particularly in how the *mvbest* file is set up. Adjusting BLM is all about how goals are met (under, precisely, overrepresented) (Ferdana, 2006). Please consult Game and Grantham (2008) for an excellent description on how BLM should be employed.

g. ***Flexible / Low Irreplaceability Units***: Areas that are not being indicated as highly irreplaceable are not unimportant for conservation; they are more flexible--as there are other planning units that contain similar biodiversity. A highly irreplaceable area may not be known locally for its biodiversity value, but may, for example, contain 100% of the occurrences of a particular target and is therefore irreplaceable for meeting the representation goal for that target. Marxan is also useful for highlighting areas that have previously been overlooked- for example where goals for many targets can be met in a spatially cohesive manner that increases the likelihood of strategy effectiveness

***h. Unexpected Results*:** Targets, goals and cost surface are all crucial to creating the best portfolios. If areas that are known to be important are not being reflected in the portfolio it may be because the targets or the cost surface are not defined in a way that distinguishes those areas as different from others. For example if no highland targets are represented in the portfolios it may be because the targets have not been stratified by elevation so those highland parts cannot be distinguished as needing representation separate from the lowland areas. The goal representation for that target will be met – but it *could* be all within lowland areas.

3.2 Marxan Input File Preparation

Suggested steps to follow when conducting a Marxan analysis are diagramed in the figure on the next page. In order to run the Marxan Tools, the user will need to prepare four basic data layers. These layers include:

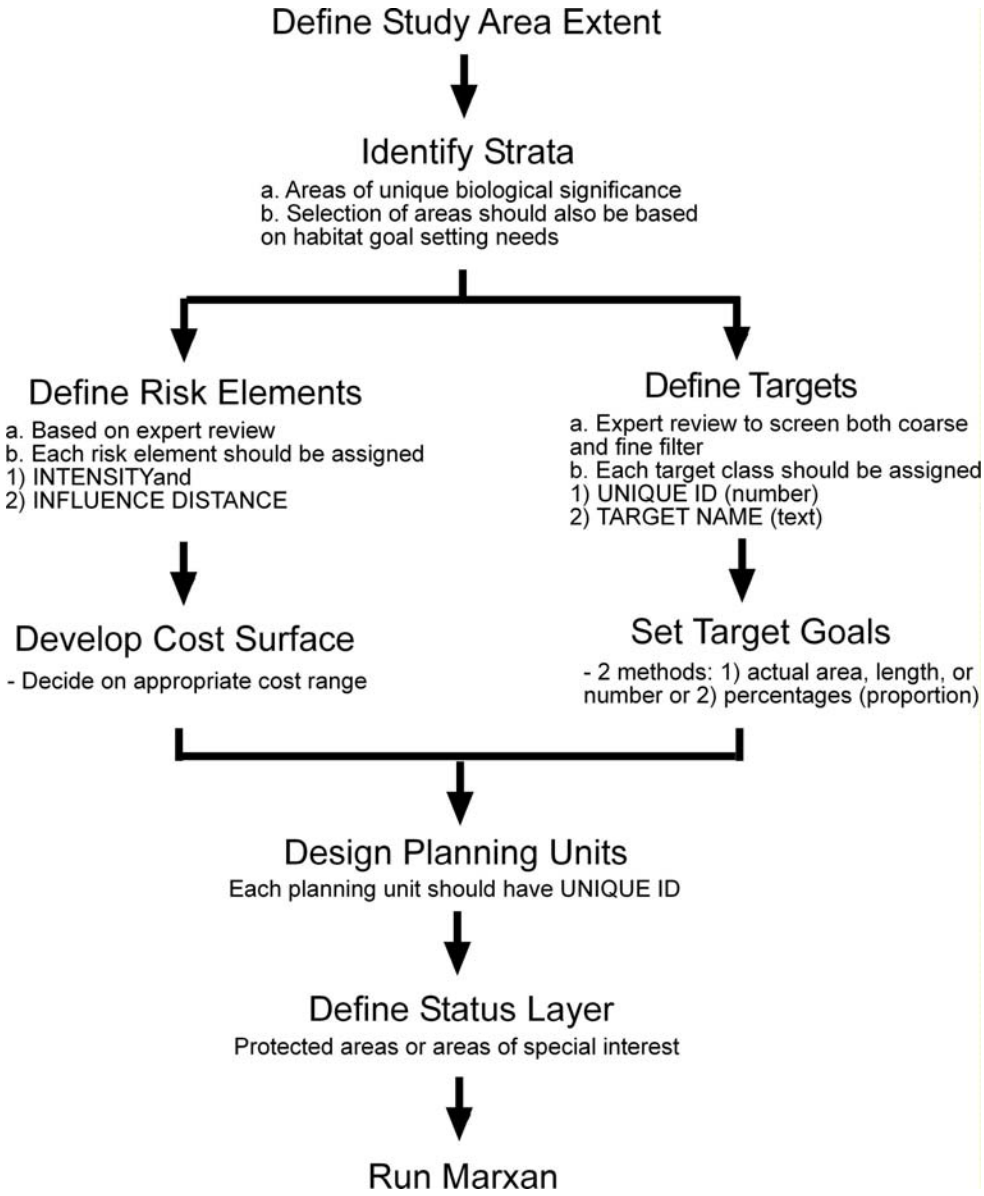
1. ***Conservation Targets***. These are the habitats or species in which conservation goals are set and can be represented as points, lines, or polygons. These features should be previously screened as potential candidate sites for meeting conservation goals that are expressed in number of occurrences (points), area (polygons) or length (lines) in map units (*i.e.* hectares). If you are using a block definition file, these goals can be expressed as percentages of the target's total units. Species penalty factors can be assigned to each target depending on how important it is for each one to reach the conservation goal. It is recommended to use the Target Prep tool when preparing your targets for the Marxan Input

Generator. This tool creates all the necessary fields and dissolves each target layer so there is one table listing for each unique target type.

2. ***Cost Surface***. This is also an optional layer, but required if the user is going to include a "Unit Cost" measure to each planning unit as indicated in the Marxan algorithm. Any number of measures may be used such as threat or suitability for conservation strategies, or a surrogate for actual cost, such as area. The Environmental Risk Surface (ERS) module can be used to create customized grids which can be summarized at the planning unit and used as a cost measure. Marxan attempts to minimize the total portfolio cost by selecting the fewest planning units with the lowest total *unit cost* needed to meet all biodiversity goals. The Marxan Input Generator allows users to assign cost to each planning unit by extracting values from a grid, using an existing planning unit field, taking the area of the planning unit, or typing in a flat cost that is assigned uniformly to all planning units.

3. ***Planning Units***. These are the units that house all the necessary information for Marxan to run and allow comparison and selection between candidate areas. Planning units can be systematic units such as hexagons; or unsystematic sub divisions of the landscape such as watersheds or administrative boundaries. The size of the units should be representative of the scale of analysis and input data resolution. For users who would like to use hexagons as planning units, the HexGen tool is available in the Marxan Tools drop-down menu.

4. ***Protected Areas or Special Interest Areas***. This is an optional layer, used in Marxan for what is called the '*Status Layer"* or *"Status Identifier*." This layer must be a polygon feature class and is required if the user plans to run a scenario where the planning units that overlap declared protected areas or special interest areas are to be "locked-in" or fixed in the reserve, thus always included in the solution. These planning units are first considered when Marxan attempts to meet the conservation goals. If goals are met within these areas, no additional planning units are selected for that particular target goal. The Marxan Input Generator permits users to specify a status layer to use when assigning the identifier. Users can also specify a field in an existing planning unit attribute table.


Define Study Area Extent
Identify Strata
a. Areas of unique biological significance
b. Selection of areas should also be based on habitat goal setting needs
Define Risk Elements
a. Based on expert review
b. Each risk element should be assigned
1) INTENSITYand
2) INFLUENCE DISTANCE
Define Targets
a. Expert review to screen both coarse and fine filter
b. Each target class should be assigned
1) UNIQUE ID (number)
2) TARGET NAME (text)
Develop Cost Surface
- Decide on appropriate cost range
Set Target Goals
- 2 methods: 1) actual area, length, or number or 2) percentages (proportion)
Design Planning Units
Each planning unit should have UNIQUE ID
Define Status Layer
Protected areas or areas of special interest
Run Marxan

Suggested steps for preparing input data and conducting a Marxan analysis. The four basic data layers include conservation targets, cost surface, planning units, and status layer.

3.2.1 Marxan Input Files

The Marxan Input Generator (MIG) can produce up to five input text files (three required and two optional files) that Marxan uses to run. These files are automatically generated using the four data layers described above and are in a tabular matrix format, a compatible input format for use with Marxan 1.8.10 and earlier versions. A new version of Marxan (2.0.2 – Marxan Optimised) was released in 2007 and is designed for larger and more complex Marxan investigations. This version is sensitive to the order of

planning unit identifiers in the Marxan planning unit by species sparse (or relational) matrix. Additionally, it can only use a sparse matrix format, and is not compatible with a tabular matrix format. Please consult the Marxan website to obtain a copy of the command line program "*convert_mtx.exe*" which converts existing Marxan tabular matrix files into a sparse matrix format, compatible with Marxan Optimised.

Since Marxan exercises may require several different runs involving the testing of multiple parameters, good file management is necessary to maintain organization during the analysis. A recommended Marxan working directory should be set up with two sub directories, one containing the input files, and another containing the output files. Three files should reside in the working root folder, including the Input Parameter File, 'input.dat', the Marxan program executable, 'Marxan.exe', and the InEdit executable. The Input Parameter File, 'input.dat', must also be stored in the same place as the Marxan program executable (see Game and Grantham, 2008).

Before we begin discussing how to use the tool to prepare the features and create these files, the user should understand the purpose and contents of each of these files. They are described by Huggins (2005):

1. ***Conservation Feature File*** (*spec.dat*)

   This file holds information about each target, including the goals and names. Only id, type and target are essential, all other variables are optional, although assigning a high (*e.g.* 10000) Species Penalty Factor (spf) will help ensure that your goals are met. If a column is missing, the default values will be used. For some columns, a value of –1 indicates either that the default is to be used or that value is given in the block definition file. The name column can contain spaces or other word separators, but any separator will be replaced by a single space. If there are any duplicate definitions, all but the last one will be ignored. The contents of this file are:

id	**Type**	**target**	**spf**	**target2**	**sepdistance**
Id of target- *must correspond to puvspr.dat file* **CRITICAL**	Looks for block definitions **DEFAULT (-1)**	Goal representation of the target **DEFAULT (-1)**	Species penalty factor for each target **DEFAULT (-1)**	Minimum clump size *optional* **DEFAULT (-1)**	Minimum separation distance *optional* **DEFAULT (-1)**

cont..

Sepnum	**name**	**targetocc**
Target number of mutually separated PUs in valid clump *optional* **DEFAULT (-1)**	Name in words can include spaces all words must start with a letter *optional* **DEFAULT (no_name)**	Number of occurrences of the target required. *optional* **DEFAULT (-1)**

The values are separated by commas. The file format looks like:

id, type, target, spf, target2, sepdistance, sepnum, name, targetocc
334,334,877676.56,10000,443,1000,2,limestone_forest,0
335,335,639282.62,63928.26,227,1000,2,alluvial_forest,0

2. ***Planning Unit File*** (*pu.dat*)

This file contains all the information related to planning units except for the distribution of targets. The column headers can include: **id**, **cost**, **status**, **xloc** and **yloc**. The **id** column is the only one that is not optional. The cost and status (ability to "lock-in') will assume a default value of 1 and 0 respectively if the columns are not present. The *xloc* and *yloc* columns are critical if there are spatial separation requirements for any of the targets (separation distance or number). These values represent the location of the planning units, and are usually the centroids of the polygons. The contents of this file are:

id	**Cost**	**status**	**xloc**	**yloc**
Unique ID for pu CRITICAL	Of each pu DEFAULT (1)	Whether pu is locked in or out of the system DEFAULT (0)	X centroids of pu CRITICAL for Separation	Y centroid of pu CRITICAL for Separation

The values are separated by commas. The file format looks like:

id, cost, status
1,2.3,0
2,4.8,0

The status of each planning unit can take one of 4 values (Default = 0):

Status	***Meaning***
0	The PU is not guaranteed to be in the initial or 'seed' reserve. However it still may be. Its chance of being included in the initial reserve is exactly the 'starting proportion' from the parameter input file.
1	The PU will be included in the 'seed' reserve or the initial reserve. It may or may not be in the final reserve.
2	The PU is fixed in the reserve. It starts in the initial reserve and cannot be removed (locked in).
3	The PU is fixed outside of the reserve. It is not included in the initial reserve and cannot be added (locked out).

3. ***Planning Unit versus Species File*** (*puvspr.dat*)

This file contains the information on the distribution of targets across the planning units. It is sometimes called the abundance file. The contents of this file are:

Species	**pu**	**amount**
Conservation target id – must be a number CRITICAL	Planning unit id CRITICAL	Number, area, or length of a conservation target CRITICAL

The values are separated by commas. The file format looks like:

species,pu,amount
26,263,535739.34
27,271,228479.37

4. ***Block Definition File*** (*block.dat*) OPTIONAL

This optional file can be used to group targets together or to use the facility that allows the goal to be set as a proportion without calculating the actual amount in area, length or number of occurrences. When using this file, the type of each target must be set in the *spec.dat* file. If no grouping is necessary, the type names can remain the same as the target id. If you are using a block definition file, it is highly recommended that you read Game and Grantham (2008) to gain a better understanding of how this file works. The contents of this file are:

type	target	target2	targetocc	sepnum	sepdistance	prop	spf
the type for which the other attributes are defined **CRITICAL**	The goal for the target of the given type. **DEFAULT (-1)**	Minimum clump size. If a clump of a number of planning units with the given target is below this size then it does not count toward the goal. **DEFAULT (-1)**	The number of occurrences of the target required. This can be used in conjunction with or instead of 'target'. **DEFAULT (-1)**	Target number of mutually separated planning units in valid clumps. **DEFAULT (-1)**	Minimum distance at which planning units holding the target are considered to be separated. **DEFAULT (-1)**	An alternative to target. This is the proportion of the total amount of the target which must be preserved. **(N/A)**	The penalty factor for that target. **DEFAULT (-1)**

The values are separated by commas. The file structure looks like:

type,target,target2,targetocc,sepnum,sepdistance,prop,spf
1,-1,-1,-1,-1,-1,0.25,10000
2,-1,-1,-1,-1,-1,0.30,10000

5. ***Boundary Length File*** (*bound.dat*) OPTIONAL

The boundary length file contains information on the boundary costs of adjacent planning units. Although this is an optional file, it is highly recommended to use since it helps set the level of fragmentation in the solution. Whereas this cost is typically the actual length of the boundary it can be modified to a 'cost' or 'effective length' value to take into account boundaries that are particularly desirable or undesirable. *Bound.dat* can be created automatically using the Marxan Input Generator. This table can have tabs or commas between the columns. If you see repeating PU ids in the bound file - these are "edge" planning units and repeating them in the *bound.dat* file helps to avoid bias in the BLM because they have shorter 'shared' boundaries. It is not necessary to specify boundary lengths for all planning units (where they are not specified, Marxan will assume there is no boundary between planning units). However any missing values within the file will prevent Marxan from running, for instance if 'id1' and

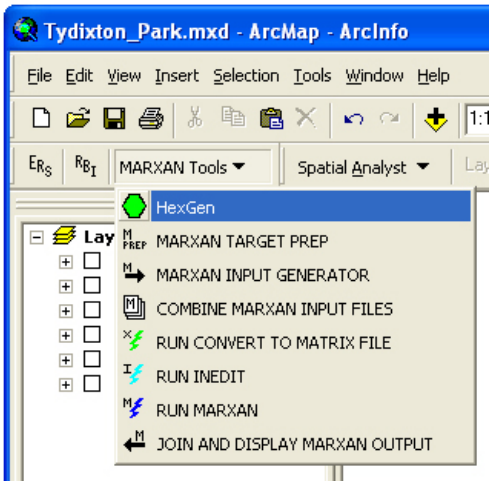
'id2' are set but no value for 'boundary' is entered (Game and Grantham, 2008). The contents of the *bound.dat* file are:

id1	id2	boundary
planning unit id CRITICAL	neighboring planning unit id or the same as id1 for an irremovable boundary CRITICAL	the boundary length CRITICAL

The values are separated by commas. The file structure looks like:

id1,id2,boundary
1,2,33453.5
1,3,334536.2

3.2.2 Generating Hexagons (HexGen)

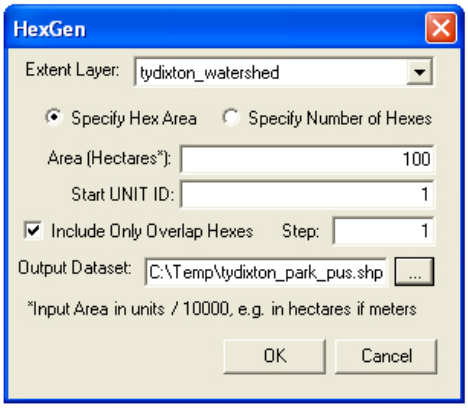


Hexagons are the most often used polygon feature that serve as planning units for Marxan analyses. Ecologists have adopted hexagons because they have more facets for connectivity and can represent a more continuous and natural surface. The HexGen tool has been designed as a simple way to generate hexagons based on a user-defined size and attribute them with customized unique IDs.

You will not need to create hexagons to run the tutorial data since they are already provided. In order to create hexagons for a new Marxan analysis, you must first decide on an extent layer which repreents your maximum extent. HexGen will use your specified extent layer to create a hexagon file that matches the spatial boundaries of your extent layer. This layer represents the total area where you would like hexagons to be created. Before you click on the HexGen button, make sure that you have the Extent Layer in your map view. Once you click on the HexGen button, follow these steps to create hexagon planning units:

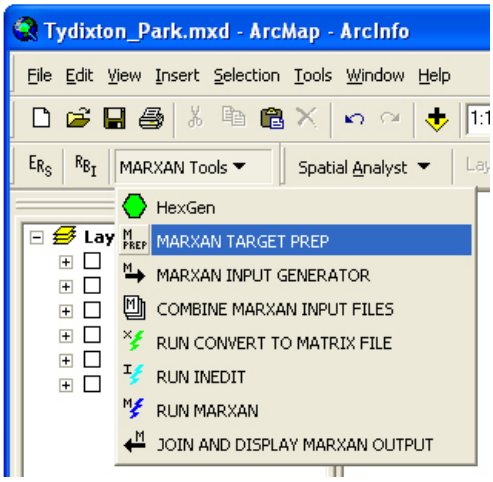
1. Specify the Extent Layer you wish to use.
2. Choose whether you would like to create the hexagons by specifying an **Area** or by specifying the **Number** of hexagons.
3. If specifying the area, enter the desired area (in hectares) for each hexagon. The default is 100ha. If specifying the number of hexagons, enter the number of hexagons you would like to create.
4. Enter a starting number for your unit IDs. The default is 1.

5. If you uncheck “Include Only Overlap Hexes” the program will return the complete set of planning units contained within the upper left and lower right coordinates of the Extent Layer. Most users leave the box checked since they only want the planning units that overlap with the features of the extent layer (e.g. watershed, targets).
6. Specify a Step interval for the calculation of the unique IDs. The default is an interval of 1 (e.g. 1,2,3,4, etc.) This is often useful when combining hexagon IDs at a later time using multiples of ten (i.e. using a step interval of 10 or 100)
7. Specify an Output File. This will be the name of the new hexagon.
8. Click OK to execute.



The process may run for quite some time, depending on the size and number of hexagons required to generate. The bottom left will report the current row of hexagons the program is processing and the status bar will show percent complete. It is best not to work on the computer or click on ArcMap while the model is executing. It takes approximately 1 minute to generate 100 hexagons (depending on the computer). Once the model is completed, a text box stating “Processing Complete” will appear and the results will be loaded into the map view.

3.2.3 Marxan Target Prep



Having reviewed the basics of Marxan and the required input files, we are now ready to start the process of preparing the conservation target data for creating Marxan input files. If the user already has their planning units (i.e. hexagons) created, the first step in using the Marxan Tools is creating and populating the target layer with the correct fields that Marxan will use to produce the input files. It is not necessary to run this step with the tutorial data provided, since these target files have already been prepared, meaning they have already been dissolved and have the required fields added and corresponding values calculated. This step is only for users who need to create Marxan files using new target data that does not already have the required fields added and values calculated.

In the PAT toolbar, you will see the “Marxan Tools” drop-down menu. Marxan Target Prep is the second tool in the menu. Before launching this tool, make sure that your

conservation target files are already loaded into the map view. To get an idea of what the Target Prep tool does, you can load any of the sample target shapefiles into the view (*target_point_species.shp, target_line_ habitats.shp*, or *target_polygon_habitats.shp*). Now, open up the attribute tables of one of the sample target shapefiles by right clicking on the layer and choosing "Open Attribute Table". You will see that in addition to the TARGET NAM and TARGET_ID fields, there are additional fields that are described in the table below. These additional fields are created using the Target Prep tool and will be used in the next step (Marxan Input Generator - MIG) to extract the field information and create the Marxan input files. More information about each of these parameters can be found in Game and Grantham (2008), Ball & Possingham (2000) and Possingham et al. (2000).

The Marxan Target Prep routine generates all the fields listed below the Target_Nam and Target_ID fields. Once populated, these fields can then be used in the Marxan Input Generator (MIG).

FIELD	*TYPE*	*DESCRIPTION*
TARGET_NAM	TEXT	*Target Name*. The name of the conservation target. This field can include spaces, but all words must start with a letter. Try to limit the length of the target name to fewer than 50 characters. A duplicate of this field will be created by the Target Prep tool and named TNAME.
TARGET_ID	INTEGER	*Target ID*. The unique ID assigned to each conservation target. The ID should be limited to no greater than 16 numbers in length. A duplicate of this field will be created by the Target Prep tool and named TID.
GOAL	FLOAT	*Target Goal*. The conservation goal for each target stated in number of occurrences (points), length (lines), or area (polygons). An alternative of defining the goal is to use the PROP field (Proportion) in the Block Definition file. Do not use both methods – choose one or the other. Make sure the units are consistent with what is specified in the MIG conversion window (default hectares and kilometers). Total field length is 15 numbers.
TYPE	INTEGER	*Target Type*. This is a user-defined number used in the block definition file. Could be the same as the Target_ID. Total field length is 16 numbers.
SPF	INTEGER	*Species Penalty Factor*. Higher numbers (*e.g.* 100,000) ensure that conservation goals are met. Can be adjusted accordingly for priority targets. Total field length is 16 numbers. Total field length is 16 numbers.
OCC	INTEGER	*Target Occurrence*. The number of occurrences of the conservation target required. This can be used in conjunction with or instead of the goal. Total field length is 16 numbers.
CLUMP	INTEGER	*Minimum Clump Size*. If a clump of a number of planning units with the given conservation target is below this size, then it does not count toward the goal.

		Total field length is 16 numbers.
SEPNUM	INTEGER	*Separation Number*. Goal number of mutually separated planning units in valid clumps. Total field length is 16 numbers.
SEPDIS	INTEGER	*Separation Distance*. Minimum distance at which planning units holding the conservation target are considered to be separated. Total field length is 16 numbers.
PROP	FLOAT	*Proportion*. An alternative to the GOAL and used in the block definition file. This is the proportion of the total amount of the conservation target which must be preserved. Expressed in 0-1. If this is specified, it is used instead of the number specified in the GOAL field. Total field length is 15 numbers.

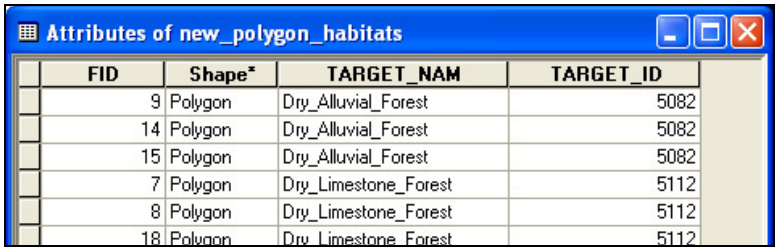
NOTE: In using the Marxan manual by Ball & Possingham (2000) please note that the authors use the term "Conservation Feature" for what we are referring to as the "Target" and they use "Target" for what we refer to as the "Goal."

Since the tutorial target data is already provided with the fields needed to extract and create the input files, there is no need to run Marxan Target Prep on them. If you would like to practice using this tool, there is a copy of the sample targets (without the last eight attributes listed in the above table) located in the *Tydixton_Park_Tutorial_Data\ Sample_Models\MARXAN\For_Target_Prep* directory that can be run through Target Prep. You can use the numbers listed in the table below to prepare these targets. It is highly recommended that you use the Target Prep routine *if you have new target features that need new Marxan input fields added and assigned values to the attribute table*. However, if you are using new target data, there are two fields that need to be manually defined before using Marxan Target Prep:

1. TARGET_NAM – A text name needs to be assigned to each point, line, or polygon that describes each unique conservation target class. This can include spaces, but all words must start with a letter (*e.g.* Semi-deciduous forest, West Indian whistling duck, etc.). When using your own data, make sure any entry in the target name field does not contain any special characters (*e.g.* ñ, ó, œ) or the program may fail. Try to limit the length of the target name to fewer than 50 characters.

2. TARGET_ID – A unique integer number must be assigned to each unique target class (*e.g.* 1,2,3 etc). Double check to make sure that you have not assigned the same ID to any of your target classes.

An example of what an attribute table should look like prior to using the Marxan Target Prep tool.



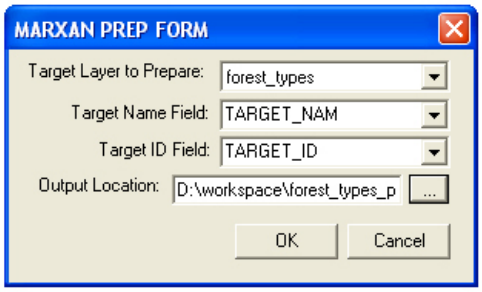
Before executing the Marxan Target Prep tool, it is a good idea to create a master table that lists all unique conservation target classes with each input field and corresponding value that will be assigned. This can be done using Excel or another type of spreadsheet application. An example of what this table should look like is found below. These values could be assigned by experts who understand the conservation needs of each target. Having this table will make it easier to use the Marxan Target Prep as you type in the field values for every target class. The assignment of these values is very important because they will drive the results of your Marxan analysis and resulting conservation assessment. Consequently, these numbers should be reviewed by conservation experts who understand both how Marxan operates and the objectives of your conservation planning exercise.

An example of a conservation target list table that should be prepared prior to running the Marxan Target Prep tool. Remember that you can use either the GOAL field or the PROP (proportion) field to define your goals, but choose one or the other. If you choose PROP, you must define a block definition file.

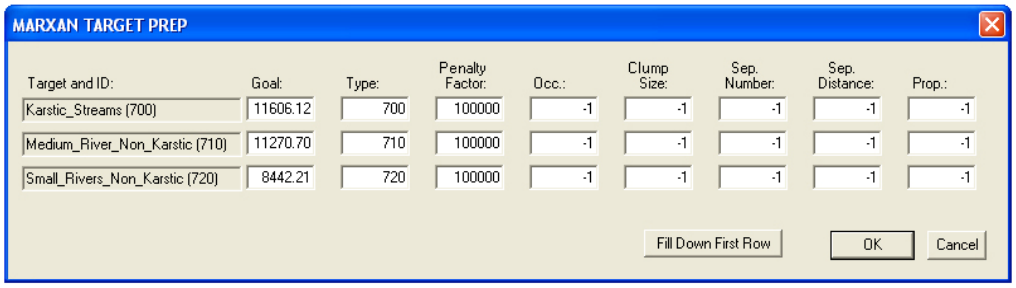
	TARGET_NAM	TARGET_ID	GOAL	TYPE	SPF	OCC	CLUMP	SEPNUM	SEPDIS	PROP*
Points	Bat_Phyllonycter	531	2.00	531	100000	-1	-1	-1	-1	0.66
	Bat_Pteronotus	524	1.00	524	100000	-1	-1	-1	-1	0.50
	Blk_Bill_Parrot	47	3.00	47	100000	-1	-1	-1	-1	0.66
	Cave	638	7.00	638	100000	-1	-1	-1	-1	0.30
	Endemic_Fish	70	2.00	70	100000	-1	-1	-1	-1	0.40
	Endemic_Turtle	80	2.00	80	100000	-1	-1	-1	-1	0.50
	Huitas	91	4.00	91	100000	-1	-1	-1	-1	0.66
	Spring	639	3.00	639	100000	-1	-1	-1	-1	0.30
Lines	Karstic_Streams	700	11606.12	700	100000	-1	-1	-1	-1	0.30
	Medium_River_Non_Karstic	710	11270.70	710	100000	-1	-1	-1	-1	0.20
	Small_Rivers_Non_Karstic	720	8442.21	720	100000	-1	-1	-1	-1	0.50
Polygons	Dry_Alluvial_Forest	5082	34.91	5082	100000	-1	-1	-1	-1	0.9
	Dry_Limestone_Forest	5112	472.05	5112	100000	-1	-1	-1	-1	0.5
	Wet_Alluvial_Forest	5132	61.39	5132	100000	-1	-1	-1	-1	0.8
	Wet_Limestone_Forest	5172	2422.14	5172	100000	-1	-1	-1	-1	0.5

*Only use the PROP (proportion) field if you are not entering exact goal amounts in the GOAL field. If using PROP, you must check the box in the Marxan Input Generator to create a block definition file.

In addition to adding the target name and ID fields, you may want to make sure that all targets to be used in the Marxan analysis have been properly screened by experts, and stratified if necessary. For example, stratification is needed if the user wants to assign different goals to the same target class that is spread across different geography settings (e.g. political boundaries, elevations or geologic zones). These stratified targets must have both a unique name and ID defined. At a minimum, the targets should have the conservation goals defined. Remember, you can use either the GOAL field to define the goals in specific units (number of occurrences, length, or area) or you can use the PROP field to set up proportions (percentages). If you use the PROP field, Marxan will calculate the appropriate target amount based on the proportion specified using a block definition file. If GOAL and PROP are both set, then the PROP variable will take precedence and the GOAL variable will be ignored in the block definition. Also note that the proportion is based on the total amount defined in the *puvspr.dat* file. Before moving on to the next step, you are also going to need your planning units (with unique IDs), the optional status layer (e.g. protected areas) if using a status flag for locking in areas, and a method for extracting the optional cost parameter (*e.g.* ERS model).



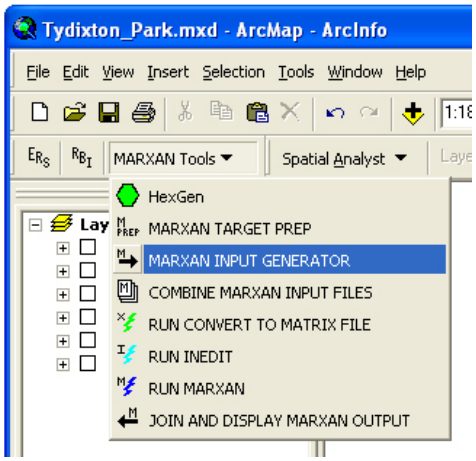
Once you click on Marxan Target Prep, a dialog box appears that asks you to select the Target Layer to Prepare and corresponding Target Name and ID Fields. You must also specify an output feature class (*e.g.* shapefile). A new feature class will be created so make sure you give it a descriptive name and have it written out to your Marxan working directory. When you are ready, click OK and the new feature class will be created containing all the additional fields and a new dialog box will appear, listing each unique target on the left side and the fields to be populated along the top. The user can now enter the values of each field using the table previously prepared. Each field has the default value already assigned but these numbers can be changed according to the conservation scenario that has been set up in the data prep table. If you have many targets that need to be assigned the same value, you can use the "Fill Down First Row" to automate the calculation of the fields using whatever values you enter in the first row. These values will be duplicated to all other rows.



When you are finished entering in all the values, recheck the numbers and press OK. Now add the new feature class to the map view and open the attribute table in order to make sure that the fields have been created and they have been populated with the correct

values. These values will be used in generate the input files that Marxan will use. You may notice that additional fields TNAME and TID were created. These are created for consistent naming conventions when using the Marxan Input Generator, which you are now ready to start in order to create the input files that Marxan will use.

3.2.3 Marxan Input Generator (MIG)



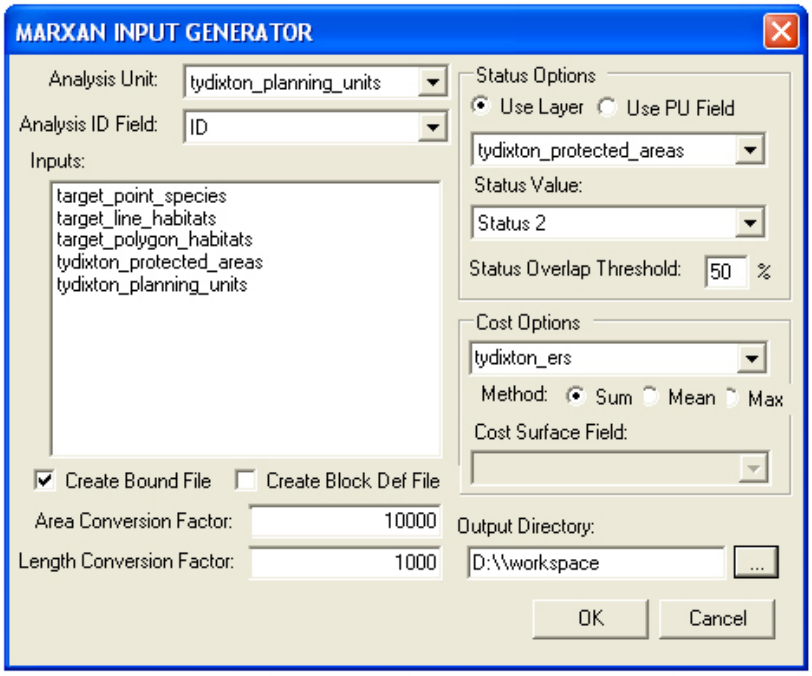
Now that the target layers have the required fields and corresponding values assigned, you are ready to use the Marxan Input Generator (MIG) to create the input files that Marxan will use. Make sure the following data layers are in your map view's table of contents before proceeding with MIG (both the status and cost layers are optional for running Marxan):

1. *Planning Units* - a polygon feature (e.g. hexagons) with unique ID assigned.
2. *Target Layers* – these must be previously run through the Target Prep tool or have all required fields with user-defined values. Targets must also be dissolved so there is one record per unique target class (Target Prep does this for you). If they are not dissolved, you will have repeating targets listed in the *spec.dat* and *block.dat* files.
3. *Status Layer* - this is an optional layer but needed if you plan on using a protected area layer or other special interest areas to "lock in" planning units. The user also has the option of locking in planning units by specifying a pre-assigned planning unit field value (0, 1, 2, 3) that is extracted and written to the *pu.dat* file.
4. *Cost Layer* – also an optional layer but needed if you are using planning unit cost values in your Marxan analysis. For defining cost, the user has the following options to choose from: a) choosing a grid (e.g. ERS model) whose planning unit mean value will be extracted and written to the *pu.dat* file; b) selecting a pre-defined cost field that already exists in the planning unit layer; c) using the area of the planning unit as cost; or d) typing in a flat cost that will be assigned to every planning unit.

Before executing MIG, remember that all input layers must be in the same projection and have the projection defined. You must also make sure you have an ArcGIS – ArcInfo Level license with ArcInfo Workstation installed. "ArcMap – ArcInfo" should appear in the title bar of your ArcGIS program. Also, check your ArcGIS Start Up menu to see if ArcInfo Workstation is listed, which indicates it has been installed. The Spatial Analyst extension must also be turned on (Tools > Extensions). Once you verify these requirements, you are now ready to launch MIG.

For this exercise, we will use the planning units (*tydixton_planning_units.shp*), all the target input files (*target_point_species.shp*, *target_line_habitats.shp*, *target_polygon_habitat.shp*), the protected areas (*tydixton_protected_areas.shp*), and the sample ERS model grid (*tydixton_ers*). When the MIG dialog box appears, you will be prompted to

enter in several fields and choose between many options that tell the program where to extract the values that it will use to create the input text files for Marxan.



Each of the items in the Marxan Input Generator menu will now be explained:

Planning Unit: Specify the planning unit feature class (polygon).

Planning ID Field: Specify the field that contains the unique ID for each planning unit.

Target Inputs: Choose all the target layers (points, lines, or polygons) that will be used to create the Marxan input files. Click on the layer to select it and click again to unselect. All layers in the current map view will appear here. You cannot use layers with the same name.

Create Bound File: Click this box is you would like to create a boundary file (*bound.dat*). You will need this file if you are going to specify a boundary length modifier (BLM) parameter in your Marxan runs.

Create Block Definition File: Click this box if you would like to create a Block Definition file (*block.dat*). If choosing this option, you need to have the following fields assigned in your target files: Type (TYPE), Species Penalty Factor (SPF), Target

Occurrence (OCC), Minimum Clump (CLUMP), Separation Number (SEPNUM), Separation Distance (SEPDIS), and Proportion (PROP).

Area/Length Conversions: The default units are specified in hectares (area) and kilometers (length). If using the area of the planning unit as cost, the cost will be reported in hectares. You should only change the default numbers if you are using units other than hectares (10000 sq m) for polygons and kilometers (1000 m) for length (*e.g.* feet). **The user should also make sure that the specified goal units are the same as the calculated units.**

Status Options Status is optional but needed if you plan on using a protected area layer or other special interest areas to "lock in" planning units. Click the "Use Layer" option for specifying a protected area polygon feature class (or special interest areas). Click the "Use PU Field" is you have already assigned a status field value (0, 1, 2, 3) in your planning unit file.

If you are using a layer to calculate the status values, choose the layer (e.g. *tydixton_protected_areas.shp*) in the drop-down menu. If you are using the PU field option, choose the correct field in the drop-down menu.

Status Value: If you are using a layer to calculate the status value, choose the status value you want assigned in the drop-down menu. For locking in planning units, the default and most common status value used is 2. The status values are:

0 The planning unit is not guaranteed to be in the initial or 'seed' reserve (or portfolio).
1 The planning unit will be included in the 'seed' reserve or the initial reserve.
2 The planning unit is fixed (or locked) in the reserve.
3 The planning unit is fixed outside (or locked out) of the reserve.

Status Overlap Threshold: Use this field to indicate the overlap threshold you would like to use for the intersection of the status layer with your planning unit layer. This threshold is used to indicate whether or not a planning unit will be chosen and assigned the specified status value. The default is set to 50%, meaning if the status layer overlaps a planning unit by at least 50%, it will be flagged with the

specified status value. You can control the percentage used by just typing in the percentage value.

Cost Options: There are four ways to assign a cost value to each planning unit: a) Specify a grid (e.g. ERS model) that is currently in your map view; b) Select "Planning Unit Area" to assign the area of the planning unit to each corresponding planning unit; c) Select "Planning Unit Field" to choose a pre-defined field in the planning unit layer that contains a cost value; and d) type in a flat cost that will be assigned to every planning unit.

Method: If you choose a grid to calculate the cost per planning unit, you must choose a zonal grid function (Sum, Mean, or Maximum) that will be used for each planning unit. The default is Sum which assigns the sum of all underlying grid cells within each planning unit as the cost for that planning unit. Mean assigns the mean value and Maximum assigns the maximum value of the cost on a per planning unit basis.

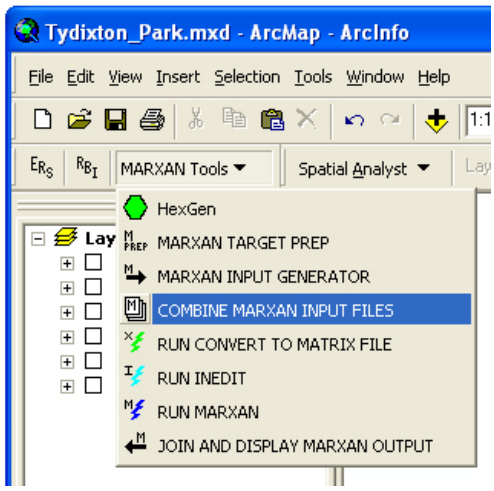
Cost Surface Field: If you choose the "Planning Unit Field" option for the cost, you must then specify the Cost Field in the drop-down menu.

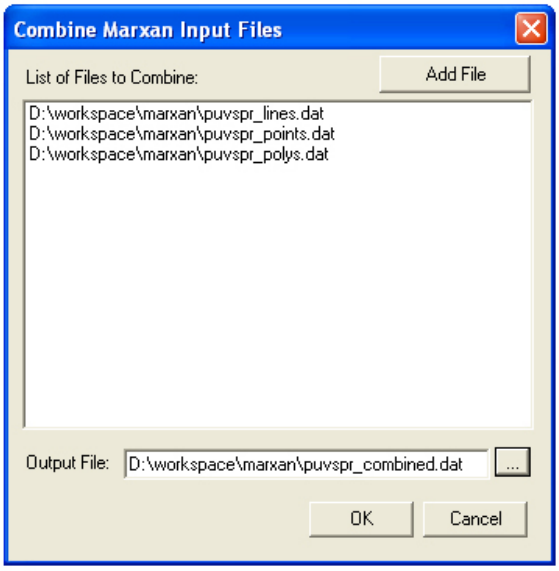
Output Directory: Specify the directory where the new Marxan output files will be written. If you are running multiple scenarios, it is helpful to assign descriptive names to the folders (e.g. *high_goals_no_lock_with_cost*). It is best not to use a directory that has a space in the path (e.g. *C:\Document and Settings\User*)

When all the parameters have been set, double check them, and then click the "OK" button. This will execute the program and create the output text files: *spec.dat, puvspr.dat, pu.dat, bound.dat* (if specified), and *block.dat* (if specified) in the specified output directory. Processing may take a long time depending on number of planning units, targets, and options specified.

3.2.4 Combining Marxan Input Files

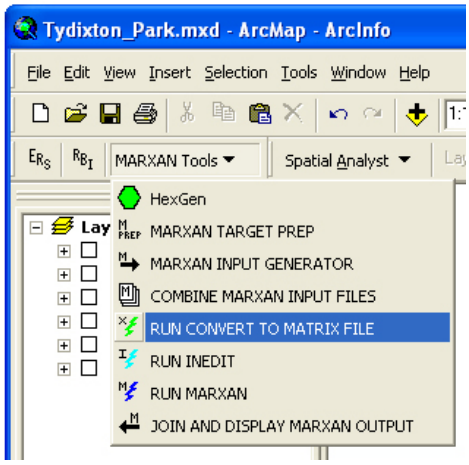
If needed, the Marxan Tools offer an easy way to combine Marxan input files. The "Combine Marxan Input Files" tool can be used to combine several input files such as *puvspr.dat* and *spec.dat* file as well as the *block.dat* file if running a block definition. This is the sixth drop down menu item on the Marxan Tools list. Just add the .dat files you want to combine, specify a new output filename, and click "OK" If you are doing this, it's important to keep track of which text files represent which feature types by including the feature type in the file name (*e.g. puvspr_points.dat*, *spec_polys.dat*). You should only have to combine multiple series of *puvspr.dat*, *spec.dat*, and *block.dat* files because the *pu.dat* and *bound.dat* files will be the same between features.





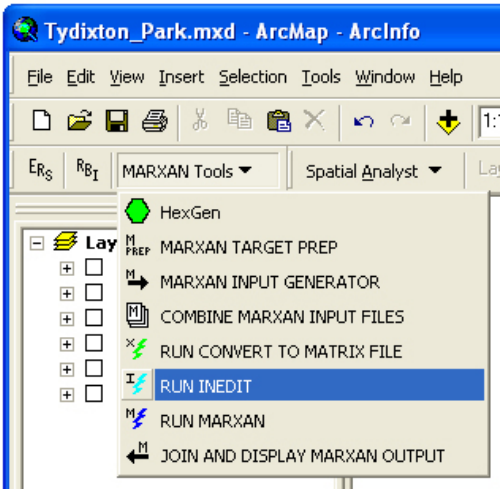
3.2.5 Run Convert to Matrix File

For those using Marxan 2.0.2 and above (Marxan Optimised), you must obtain a copy of the "convert_mtx.exe" executable available on the Marxan website in order to converts existing Marxan matrix files into a format compatible with Marxan Optimised. These versions of Marxan are sensitive to the order of planning unit identifiers in the Marxan planning unit by species sparse (or relational) matrix and can only use a sparse matrix, which is not compatible with a tabular matrix. Once the executable is available on a local hard drive, users can use this tool to point to the executable and convert the tabular (horizontal) matrix into a sparse (relational or vertical) matrix.



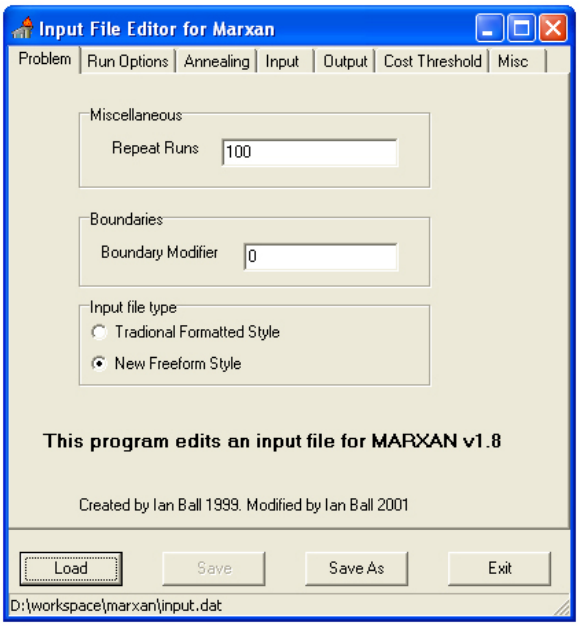
3.3 Setting up the input.dat file with Inedit and Executing Marxan

3.3.1 Running Inedit



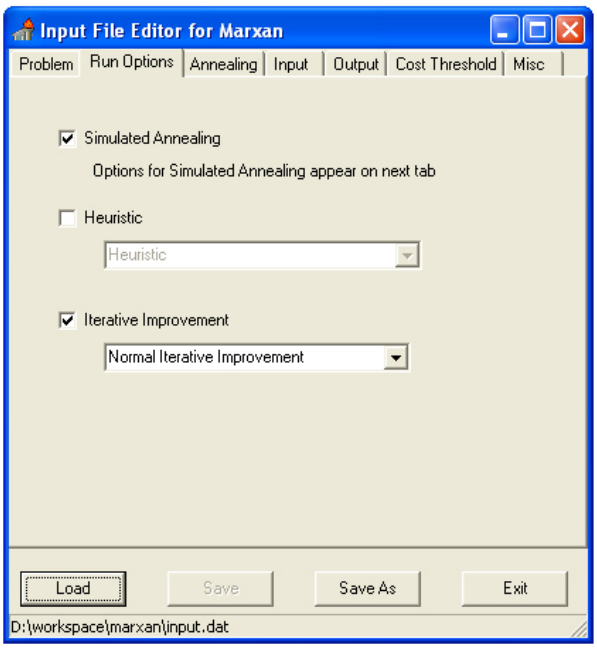
You are now ready to create the *input.dat* file using the Inedit.exe program, the "Input File Editor for Marxan" program created by Ian Ball. The *input.dat* file will be read into Marxan, using the user-defined parameters. This section was adapted from Huggins (2005) and will walk you through the process of creating the *input.dat* file using the text files you just created with the MIG tool.

You have the option of running the Inedit.exe and Marxan.exe outside of ArcGIS, or the Marxan Tools provide an option to specify the location of the executable files using the "Run Inedit" and "Run Marxan" options on the drop-down menu. If you have not already specified the location of these executable files, the tool will prompt you to locate them. You must do this for every new ArcGIS session. It is useful to have the Inedit executable file located inside each Marxan directory you plan to use. If you do not already have an *input.dat* file in the same directory, an error will appear. Click OK and disregard the error. We will be creating this file so the next time you launch Inedit, the error will not appear.

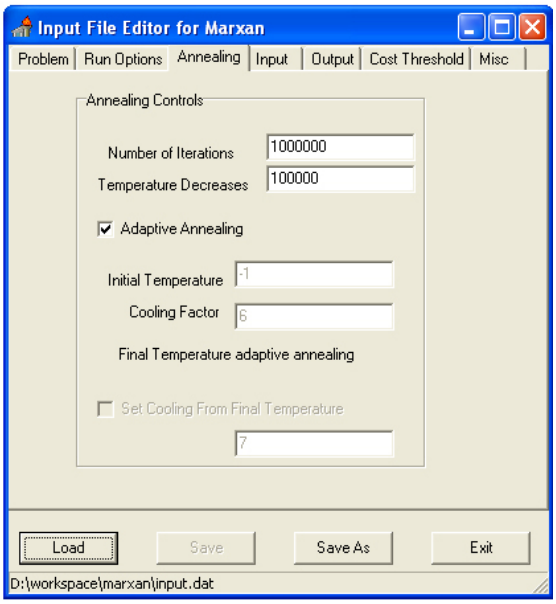


1. The first screen is to set the **Problem**. The options consist of the number of runs to be performed and the level of clustering desired. Type **100** runs in the first 'Miscellaneous – Repeat Runs' box. 200 runs are also often used, but take twice as long to run. 100 runs are used here due to time constraints. Type **0** into the **Boundary Modifier** box as shown here. The type of input file to be used is the **New Freeform Style**.

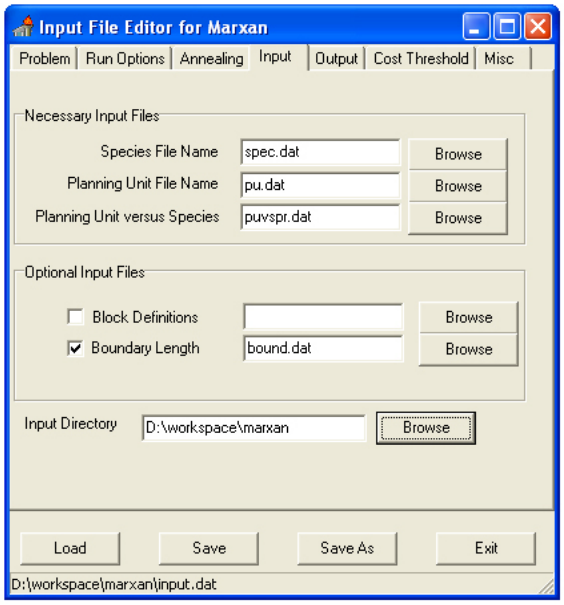
2. Click onto the **Run Options** tab positioned second in line at the top of the Inedit screen. You should see options for the type of algorithm to be run by Marxan. Click the box for **Simulated Annealing** and for **Iterative Improvement.** Click on the **Save** button at the bottom of the Inedit screen.



3. Click on the **Annealing** tab positioned third from the left at the top of the Inedit screen. The default value for the annealing controls should be used. These are 1000000 (six zeros) **Iterations** of the algorithm with 10000 (four zeros) **Temperature Decreases**, used with **Adaptive Annealing**.

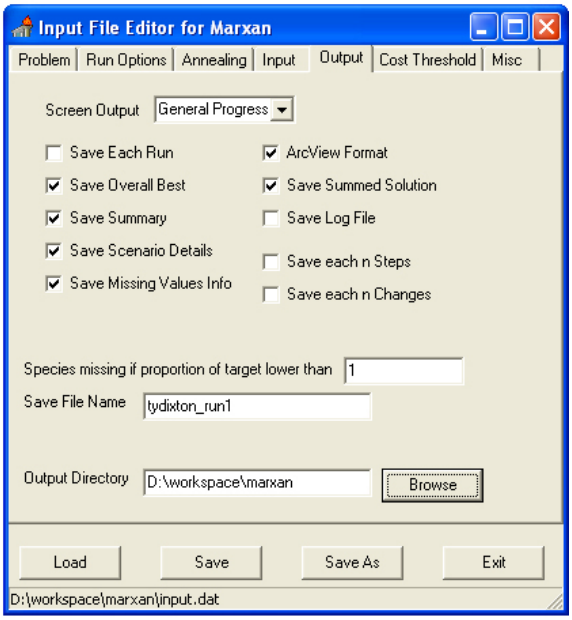


4. Click on the **Input** tab of the Inedit screen. The input folder must be set first before navigating to the input files. Click on the input directory browse button towards the bottom of the screen and navigate to the output folder where your new input files are located.



The *input.dat* files can then be specified by clicking on the browse button next to each input file name and navigating to the corresponding file. The necessary files are as follows; Species File Name = ***spec.dat***, Planning Unit File Name = ***pu.dat*** and the Planning Unit versus Species = ***puvspr.dat***. We will not run a block definition this time so un-check the '**Block Definitions**' file locator under the 'Optional Input Files'. Click on the box to check the boundary length file option and browse to the ***bound.dat*** file. Click on the **Save** button to save the information to the ***input.dat*** file.

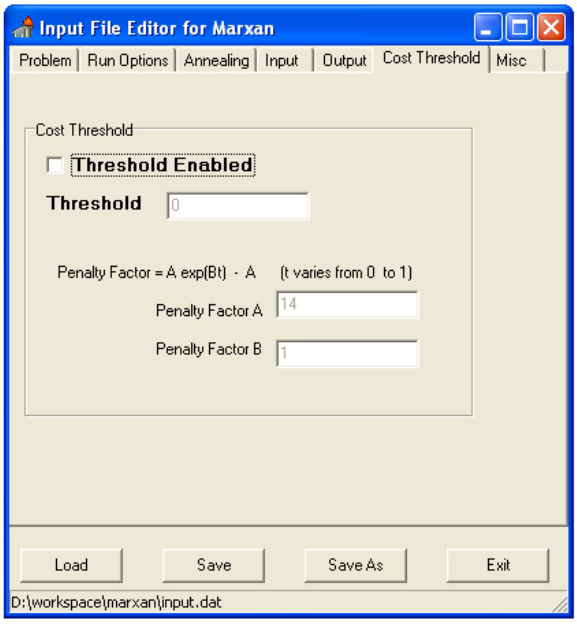
5. Click on the **Output** tab of the Inedit screen. This screen allows the user to specify the output files required to analyze the results. Screen Output: **General Progress** gives a good idea how the algorithm is running. Click on the output files shown in the graphic to the right. Saving each run has minimal use and produces large amount of text files. The most important files are the **Overall Best**, the **Summary** and the **Summed Solution**. Type **1** in the 'Species missing if proportion of target lower than' box. This option is to allow Marxan to consider a target to have met its goal if it is very close. This can be useful if the goals are high and the cost to meet the last small part of the goal would be very 'expensive'.



The text within the box '**Save File Name**' will prefix all the output text files. Enter a descriptive name that will help you remember the specific run that the output files refer to (*e.g.* tydixton__100runs). In the **Output Directory**, enter the same folder path where you input files exist. It's best to keep all the files associated with the same run together.

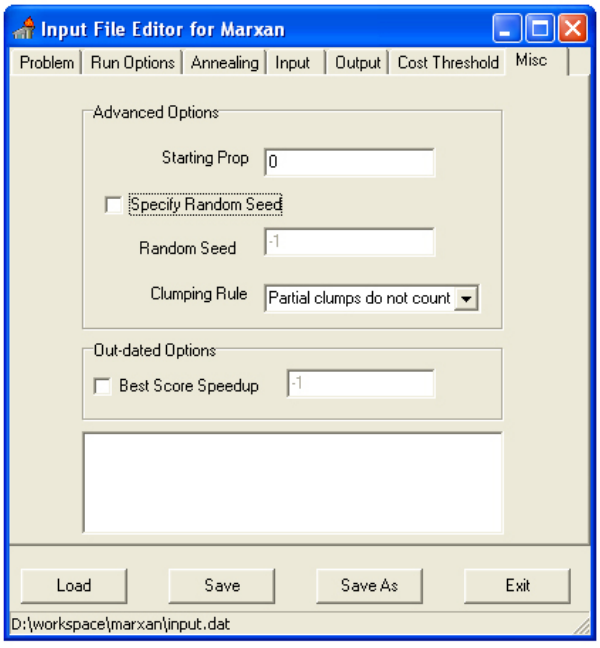
This is the folder where the output files will be written by Marxan. Click the '**Save**' button to save the information to the ***input.dat*** file.

6. Click on the **Cost Threshold** tab on the Inedit screen. This screen should be left with the default values and the **Threshold Enabled** box unchecked as shown.



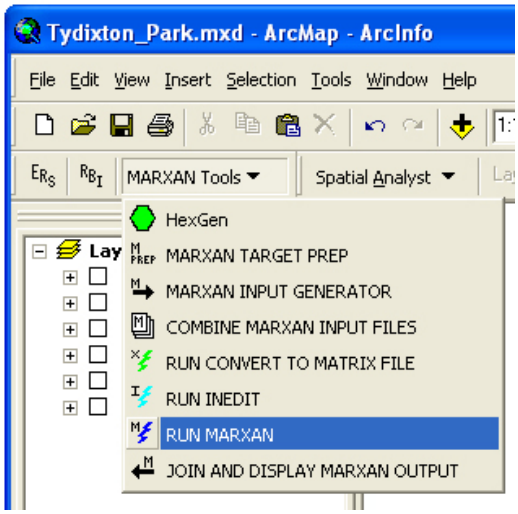
7. Click on the **Misc** tab on the Inedit screen. This tab allows the user to specify a starting proportion of the planning units to be in the random starting portfolio. Type **0** into this box, and uncheck the **Specify Random Seed** box. The **Clumping Rule** and the **Best Score Speed Up** option will not be in use, so they can be left unchecked as shown in the graphic. Click on the '**Save**' button to update and save the *input.dat* file. This is the file that the Marxan.exe file will read. Exit the Inedit program by clicking on the '**Exit**' button.

Now you are ready to run Marxan.



3.3.2 Running Marxan

The *Marxan.exe* should be located in the same directory as your *input.dat* file. You can launch the Marxan.exe either through the Marxan Tools drop-down menu, specifying its location, or double clicking on the Marxan.exe from within windows explorer. A screen will appear to show the progress of the algorithm. The program should run for less than 5 minutes using the tutorial data, depending on the speed and available resources of your computer.



Error messages at the beginning showing warnings about *blockdefname* and *highdata* are normal. For the tutorial there should be 113 planning units, 22 species (targets), 334 boundaries and 85 conservation features read by the algorithm.

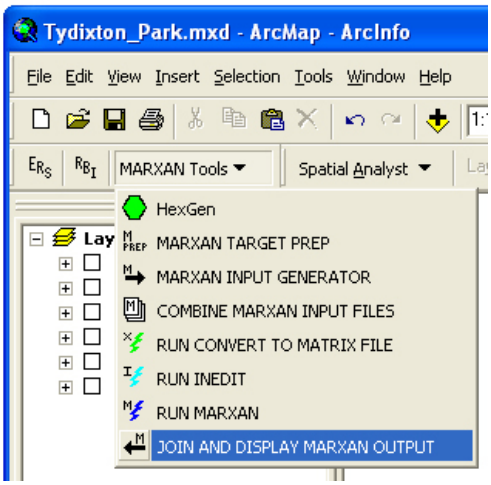
The information screen will then print any further information. This can help identify any problems with the input files. A message will appear here if one or more targets are already adequately represented in a portfolio (*e.g.* if locking in any planning units at the beginning of the run), or if there are problems with the formatting of the *dat* files causing them to be unreadable. If this occurs, check the formatting, including the placing of commas, check all field headings are lowercase (no capital letters can appear in the headings).

When the program has finished running, press <Enter> to close the window. Marxan writes the results to text files that can be found in the specified output folder. These files can be viewed using Notepad or Excel. If you decide to change the settings in the feature attribute tables, it is beast to reextract the values and create new input files in a new directory.

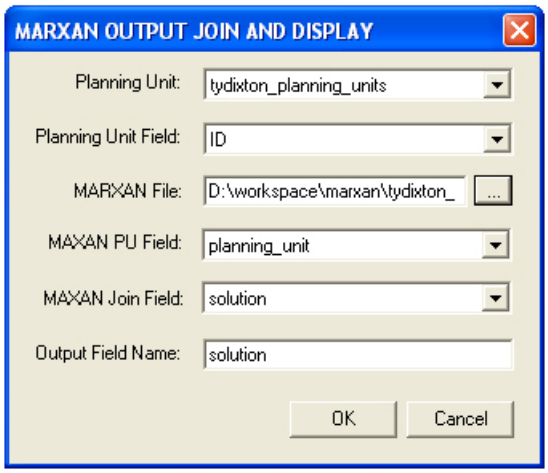
D:\workspace\marxan\Marxan.exe
Marxan v 1.8.10
Marine Reserve Design via Annealing
Coded by Ian Ball
Writen by Ian Ball and Hugh Possingham
iball@antdiv.gov.au
hpossing@zen.uq.edu.au
WARNING unable to find HIGHDATA, default value will be used.
Entering in the data files
There are 11379 Planning units.
11379 Planning Unit names read in
19 species read in
34132 boundaries entered
65727 conservation values read
Reading in the Block Definition File
Time passed so far is 1 secs
Pre-processing Section.
2 species cannot meet targets.
Run 1 Using Calculated Tinit = 258917208.0811 Tcool = 0.99980110
Creating the initial reserve
InitValue 285925367842.8 Cost 1075.0 PUs 71 Boundary 216.0 Missing 19 Shortfall 3031522.65 Penalty 285925361367.8
Value 285925367842.8 Cost 1075.0 PUs 71 Boundary 216.0 Missing 19 Shortfall 3031522.65 Penalty 285925361367.792175
Annealing Completed Time passed so far is 49 secs
Value 182058.9 Cost 82358.9 PUs 4516 Boundary 3988.0 Missing 1 Shortfall 0.00 Penalty -0.0
Value 182058.9 Cost 82358.9 PUs 4516 Boundary 3988.0 Missing 1 Shortfall 0.00 Penalty -0.001095
Best:Value 182058.9 Cost 82358.9 PUs 4516 Boundary 3988.0 Missing 0 Shortfall 0.00 Penalty 0.0
Value 182058.9 Cost 82358.9 PUs 4516 Boundary 3988.0 Missing 0 Shortfall 0.00 Penalty 0.000000
Value 182058.9 Cost 82358.9 PUs 4516 Boundary 3988.0 Missing 0 Shortfall 0.00 Penalty 0.0
Value 182058.9 Cost 82358.9 PUs 4516 Boundary 3988.0 Missing 0 Shortfall 0.00 Penalty 0.000000
Run 2 Using Calculated Tinit = 226358132.6140 Tcool = 0.99981466
Creating the initial reserve
InitValue 285925367842.8 Cost 1075.0 PUs 71 Boundary 216.0 Missing 19 Shortfall 3031522.65 Penalty 285925361367.8
Value 285925367842.8 Cost 1075.0 PUs 71 Boundary 216.0 Missing 19 Shortfall 3031522.65 Penalty 285925361367.792175

3.4 Joining and Displaying Marxan Output

3.4.2 Joining Marxan Output Files



Now let's view the results using ArcMap by joining the new output Marxan files "*run_name*_best.txt" and "*run_name*_ssoln.txt" with the planning units ID. The planning unit ID acts as the common key to join these text files. To do this, click on the Join and Display Marxan Output menu item. This will bring up a new dialog box and you must specify the Planning Unit feature class, the Planning Unit ID field, the Marxan output file you wish to join, and the name of the join field, and the new output field which will store the new information. You will have to run this routine twice: once for the "*run_name*_best.txt" (Field Name should be SOLUTION), and another time for the "*run_name*_ ssoln.txt" (Field Name should be NUMBER). The SOLUTION refers to which planning units actually made it into the final portfolio. Planning units assigned the number 1 are included, those assigned a 0 are excluded. The NUMBER refers to how many times the planning unit was chosen based on the total numbers of runs that were specified. For example, if you entered 200 runs, and a planning unit's NUMBER field is assigned 153, then that planning unit was chosen 153 times out of the 200 runs. The NUMBER is also often referred to as a level of *irreplaceability*.



When you join the planning unit ID field with the text files, a new field is created in the planning unit feature class and the results are displayed in the map view. Planning units with a value of zero (0) are excluded from the classification. You may want to make a copy of the planning unit file for each scenario that was executed (*e.g. cost_palock, cost_nopalock, nocost_palock, nocost_nopalock*).

3.4.2 Displaying and Analyzing Marxan Results

You can now analyze each of the six types of output files that Marxan has produced during the run (Ball & Possingham, 2000):

1. **Solutions for each run** (*e.g. tydixton_run1_best.txt*) A file is produced for each run of the algorithm containing the planning units which constitute the final solution produced by that algorithm. The file consists of a list of planning unit ids which constitute the portfolio.

2. **Missing value information for each run** (*e.g. tydixton_run1_mvbest.txt*) This file contains information on how well the final portfolio from each run and also the best run did with regard to meeting the conservation target's goal. The final column is the actual amount of that target which appears in the portfolio. A sample listing is shown in Table 3.

3. **Best solutions of all runs** (*e.g. tydixton_run1_best.txt*) It contains the best solution from a set of runs. The file consists of a list of planning unit ids which constitute the portfolio.

4. **Summary information** (*e.g. tydixton_run1_sum.txt*) It contains summary information on each run with a header line which describes what is in each line.

5. **Scenario details** (*e.g. tydixton_run1_sen.dat*) It contains a documented list of the options that made up that scenario.

6. **Summed solutions over all runs** (*e.g. tydixton_run1_ssoln.txt*) Each line has the id number of a planning unit and the number of times that that planning unit was involved in a solution.

Now let's examine a few of these new Marxan output files. Double click on **tydixton_100runs_mvbest.txt** (or whatever file name it was given) in windows explorer to open it within Notepad (choose Notepad from the list of programs if it does not open automatically). It may be easier to view the data in columns by importing the output text files as comma-delimited in Excel. Check to see if each target met its goal (remember that the goal is called "target" by Marxan) by looking at the last column in the 'mvbest' file. An example of the mvbest sample output is found in Table 3. In this table, the target "Wet_Alluvial_Forest" did not meet the goal. You can determine how much more is needed by looking at the "Amount Held" field and comparing it to the Target field (which is the Goal). A high Species Penalty Factors (SPF) value (*e.g.* 100000 - penalty for not representing targets to the set goals) for all targets will raise the relative importance of the target representation in comparison to overall portfolio cost (a combination of boundary length and modifier, cost per planning unit and SPF), thereby forcing the representation goals to be met (Huggins, 2005). Also check **tydixton_100runs_sum.txt** to view the score of each run.

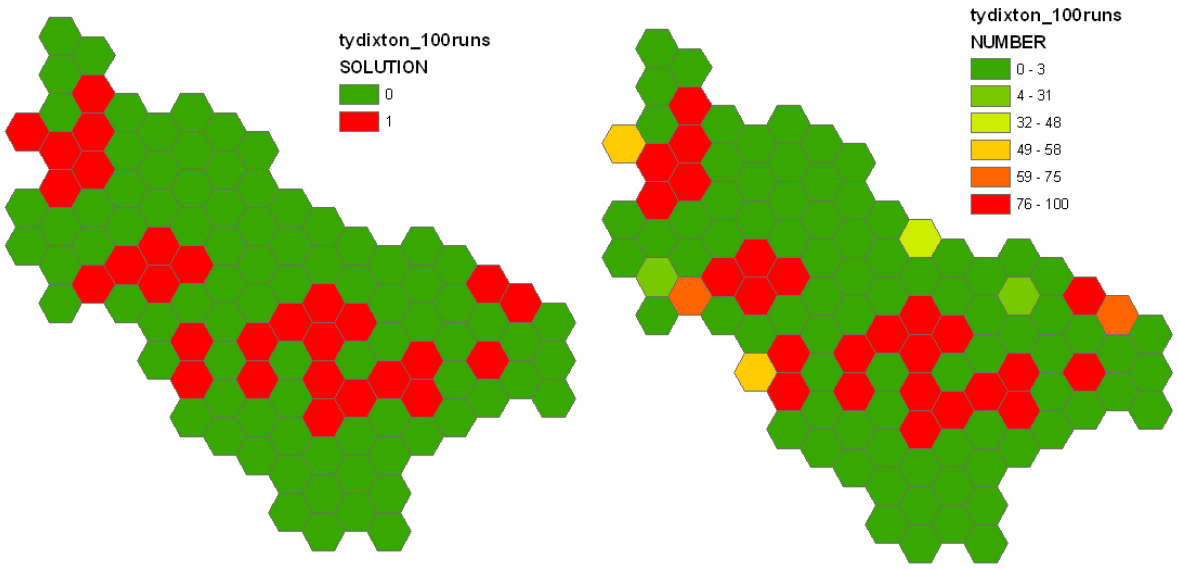


Figure 12. Results of joining the *tydixton_100runs_best.txt (*"Solution" field) and *tydixton_100runs_ssoln.txt* ("Number" field) with the planning unit "ID" field. These are the planning units that constitute the portfolio (Solution) that meet all the goals and the number of times that that planning unit was chosen based on 100 runs.

Table 3. Analysis of Marxan output *tydixton_100runs_mvbest.txt* which lists several model outcomes by feature including amount of the target that was held and if the goal was met. Remember the term "Target" in this output is the same as the conservation goal.

Conservation Feature	Feature Name	Target	Amount Held	Occurrence Target	Occurrences Held	Separation Target	Separation Achieved	Target Met
5172	Wet_Limestone_Forest	2422.14	2440.04	0	21	0	0	yes
5132	Wet_Alluvial_Forest	61.39	60.76	0	4	0	0	no
5112	Dry_Limestone_Forest	472.05	549.85	0	6	0	0	yes
5082	Dry_Alluvial_Forest	34.91	38.48	0	3	0	0	yes
639	Spring	3	3	0	3	0	0	yes
91	Huitas	4	4	0	4	0	0	yes
80	Endemic_Turtle	2	2	0	2	0	0	yes
70	Endemic_Fish	2	2	0	1	0	0	yes
638	Cave	7	13	0	7	0	0	yes
47	Blk_Bill_Parrot	3	3	0	3	0	0	yes
524	Bat_Pteronotus	1	1	0	1	0	0	yes
531	Bat_Phyllonycter	2	2	0	2	0	0	yes
720	Small_Rivers_Non_Karstic	8442.21	9570.59	0	6	0	0	yes
710	Medium_River_Non_Karstic	11270.7	12538.22	0	12	0	0	yes
700	Karstic_Streams	11606.12	18360.14	0	8	0	0	yes

References

Araújo, M.B., P.H. Williams and A. Turner. 2002. "A sequential approach to minimize threats within selected conservation areas." Biodiversity and Conservation 11: 1011-1024.

Ball, I. R. and H. P. Possingham, (2000) MARXAN (V1.8.2): Marine Reserve Design Using Spatially Explicit Annealing, a Manual

Convention on Biological Diversity, 2001, Global Biodiversity Outlook, CBD, Montreal, Canada

Dudley, N and J. Parrish, 2006, Closing the Gap. Creating ecologically representative protected area systems: A guide to conducting gap assessments of protected area systems for the Convention on Biological Diversity, Secretariat of the Convention on Biological Diversity, Montreal, Technical Series no. 24, vi + 108 pages.

Ervin, J. and J. Parrish. 2006. "Toward a Framework for Conducting Ecoregional Threats Assessments." In: Monitoring Science and Technology Symposium: Unifying Knowledge for Sustainability in the Western Hemisphere Proceedings, C. Aguirre-Bravo, J. P. Pellicane, P. D. Burns, and S. Draggan, eds. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station. p. 105-112

Ferdana, Z., 2006, Personal communication.

Game, E. T. and H. S. Grantham. (2008). Marxan User Manual: For Marxan version 1.8.10. University of Queensland, St. Lucia, Queensland, Australia, and Pacific Marine Analysis and Research Association, Vancouver, British Columbia, Canada

Huggins, A., 2006, Personal communication.

Huggins, A., 2005, Marxan Conservation Planning Decision Support Software Tutorial for GIS Beginners. The Nature Conservancy. 18 pages.

Huggins, A.E., S. Keel, P. Kramer, F. Núñez, S. Schill, R. Jeo, A. Chatwin, K. Thurlow, M. McPearson, M. Libby, R. Tingey, M. Palmer and R. Seybert 2007. Biodiversity Conservation Assessment of the Insular Caribbean Using the Caribbean Decision Support System, Technical Report, The Nature Conservancy.

Pressey, R.L. & Logan, V.S., 1998, Size of selection units for future reserves and its influence on actual vs. targeted representation of features: a case study in Western New south Wales. Biological Conservation, 85, 305-319.

McPherson, M, S. Schill, G. Raber, K. John, N. Zenny, K. Thurlow, and A. Sutton, 2008 "GIS-Based Modeling of Environmental Risk Surfaces (ERS) for Conservation Planning in Jamaica," Journal of Conservation Planning, Vol 4:60-89

Possingham, H. P., I. R. Ball and S. Andelman (2000) Mathematical methods for identifying representative reserve networks. In: S. Ferson and M. Burgman (eds) Quantitative methods for conservation biology. Springer-Verlag, New York, pp. 291-305.

Schill, S., 2005, Coastal Transport Modeling Using HYCOM Data, Proceedings of the 8th International Conference on Remote Sensing for Marine and Coastal Environments, May 17-19, Halifax, Nova Scotia.

Theobald, D.M. 2003. "Targeting conservation action through assessment of protection and exurban threats." Conservation Biology 17: 1624-1637.

The Nature Conservancy, 2006, South America Threat Assessment, Arlington, VA.

The Nature Conservancy, 2005, A Regional Conservation Assessment for the Greater Caribbean Basin, Arlington, VA.

The Nature Conservancy, 2003, Alaska-Yukon Arctic Ecoregional Assessment.

Zenny, N., 2006, Technical Summary of the Jamaica Ecoregional Planning (JERP) Marine Analysis, The Nature Conservancy, Kingston, Jamaica, 16 pages.